CENTRO DE CAPACITACIÓN
ANUIES
VALLE DE BRAVO

MANUAL TÉCNICO DEL SISTEMA DE ADMINISTRACIÓN DEL CENTRO DE CAPACITACIÓN ANUIES–VALLE DE BRAVO

INTRODUCCIÓN

El Sistema de Administración del Centro de Capacitación ANUIES-Valle de Bravo, tiene como objetivo, consultar, agregar, modificar, o eliminar información de los productos y servicios que ofrece el C.C. ANUIES-V.B, además de realizar funciones académicas en conjunto con el sistema Moodle Versión 1.9.x y versión 2.2.x. Dichas funciones se describen de forma breve como:

1. Guardar información de clientes, productos y servicios ofrecidos en los eventos del C.C. ANUIES-V.B.
2. Control de eventos del C.C. ANUIES-V.B.
3. Altas de cursos y sus categorías reflejándolas en Moodle, asi como los grupos que contiene cada curso.
4. Altas de instructores o profesores que se reflejen en Moodle, así como el control de gastos por honorarios de cada uno.
5. Alta de Alumnos que se reflejan en Moodle y con capacidad de sincronizar alumnos dados de alta desde Moodle y reflejarlos a su vez en el sistema
6. Generación de reportes tanto administrativos como académicos.
7. Generación de facturas electrónicas con el proveedor TRALIX usando las reglas de negocio de Tralix.
8. Pagos electrónicos y promoción de cursos via el proveedor WebPay

El Sistema y se desarrolló de forma modular, actualizable y con una base de datos normalizada y relacional, de tal forma que sea fácil de actualizar y, en su caso, migrar a otros servidores.

El sistema permite crear perfiles de usuario de manera flexible y con base en privilegios para garantizar que los responsables de cierta información sean quienes la actualicen y administren.

El sistema está desarrollado para funcionar correctamente en los navegadores más populares y actuales; permitiendo que el mismo sea consultado desde dispositivos y navegadores de versiones anteriores.

Toda la tecnología que se utilizó para el desarrollo del sistema y el sitio es OpenSource (No requiere de licencias adicionales).

Para entender el funcionamiento del sistema es necesario conocer el flujo de operación del mismo que se describe a continuación.

La operación del sistema se divide en cinco categorías:

1. Sistema.
2. Catálogos.
3. Eventos.
4. Sección Académica.
5. Reportes.

1. Sistema

Esta sección se compone de un solo módulo, que es el de administración de usuarios y sus privilegios.

2. Catálogos

Esta categoría abarca los catálogos con los que cuenta el C.C. ANUIES-V.B., de los productos y/o servicios disponibles y su administración a lo largo del sistema.

3. Eventos

En esta categoría se podrán administrar los eventos desde una simple cotización o borrador, hasta el seguimiento del evento que comprende renta de equipo, fotocopias, cafetería, hospedaje más gastos adicionales.

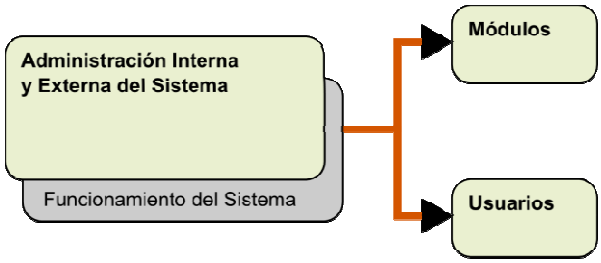
4. Sección Académica

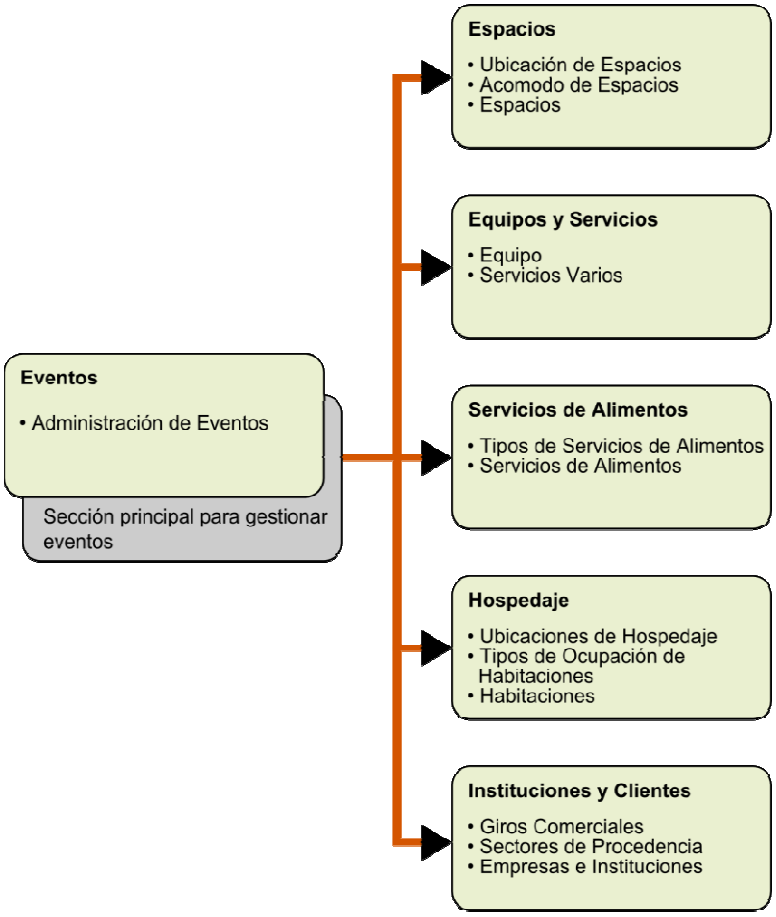
En la categoría de Sección Académica, toda la información que se agregue, modifique o elimine, se ve reflejada en el sistema Moodle ya sea en la versión 1.9.x y 2.2.x. o ambas versiones si así se configurara. Esta información abarca; cursos, grupos, instructores y alumnos.

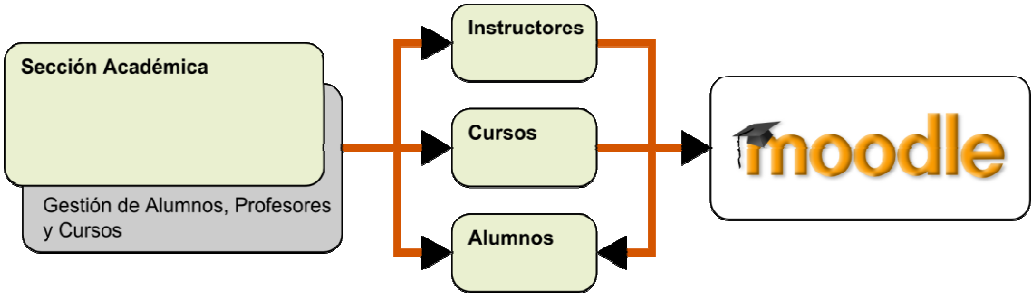
5. Reportes

En esta categoría se generan reportes de la parte académica en formatos PDF y Excel, además de la generación de constancias o reconocimientos a alumnos.

Diagrama de Bloques General




Espacios
• Ubicación de Espacios
• Acomodo de Espacios
• Espacios
Equipos y Servicios
• Equipo
• Servicios Varios
Eventos
• Administración de Eventos
Sección principal para gestionar eventos
Servicios de Alimentos
• Tipos de Servicios de Alimentos
• Servicios de Alimentos
Hospedaje
• Ubicaciones de Hospedaje
• Tipos de Ocupación de Habitaciones
• Habitaciones
Instituciones y Clientes
• Giros Comerciales
• Sectores de Procedencia
• Empresas e Instituciones


Sección Académica
Gestión de Alumnos, Profesores y Cursos
Instructores
Cursos
Alumnos
moodle

REQUERIMIENTOS DEL SISTEMA

Hardware

SERVIDOR

Considerando un máximo de 100 usuarios conectados al sistema de administración simultáneamente se recomiendas las siguientes características mínimas:

Procesador:	Intel X86, AMD64 o compatible mínimo a 2Ghz
Memoria RAM:	1 Gb
Disco Duro:	10 Gb
Enlace de ancho de banda:	Mínimo 2 Mbps
Sistemas operativos:	LINUX, FreeBSD, OpenBSD, MACOS,WINDOWS
Sistemas operativos recomendados para su operación	LINUX u OpenBSD.

NOTA: Solo se están considerando las características para el uso del sistema de control del C.C. ANUIES-V.B, las características a usar para Moodle y su funcionamiento deben de ser analizadas por parte de la ANUIES.

CLIENTE

Del lado del cliente para utilizar el sistema se tienen los siguientes requerimientos:

Procesador:	Intel X86, AMD64 o compatible mínimo a 1.2Ghz
Memoria RAM:	Mínimo 512 Mb RAM
Internet con ancho de banda:	Mínimo 128 kbps
Sistema Operativo	Todo sistema que utilice un navegador que respete estándares de HTML aprobados por el W3C y que cuente con interprete de JavaScript en su versión estándar

Software

En el servidor se requieren las siguientes características:

Servidor protocolo HTTP	Apache 1.3 o superior
Programación base:	PHP 5.3.10 o superior
Librerías instaladas al PHP	APC 3.1.2 o superior GD 2 o superior HASH MySQLi
Base de datos:	MySQL 5.0 o superior

Del lado del cliente se requiere un sistema operativo instalado, que pueda ejecutar cualquiera de los siguientes navegadores: Mozilla Firefox 3.5 o superior, Chrome 7 o superior, Internet Explorer en sus versiones 6, 7, 8; Opera 10 o superior y Safari.

TECNOLOGÍAS USADAS POR EL SISTEMA

- **Servidor**

  Programación
  El sistema está programado en PHP (http://www.php.net) basado en la versión 5.2 con las extensiones:

  Hash Message Digest Framework: Se utiliza SHA512 con una salt de 15 caracteres para la encripción de contraseñas, utilizando la librería urandom noise generator integrada dentro de la distribución del sistema operativo OpenBSD.

  PHP pecl APC (Alternative PHP cache) Librería para optimizar el cache de PHP y el uso de las conexiones. En el sistema se usa para monitorear la subida de archivos al servidor.

  MySQLi (MySQL Improved) Librería de conexión a base de datos que añade características de seguridad entre las conexiones de PHP a la base de datos MySQL.

  Manejador de base de datos
  MySQL (http://www.mysql.com) basado en la versión 5.

  Procesamiento de imágenes:
  El sistema manipula imágenes utilizando las librerías del GD.

- **Cliente**

  Contenido
  HTML 4.01 atendiendo el estándar ISO/IEC 15445.

Diseño

HTML 4.01 (W3C Recommendation 24-12-1999).
CSS2.1 (W3C Recommendation 07-06-2011).

Funcionamiento:

En el caso del sitio se desarrolló una librería en JavaScript no intrusivo (Sandox.js) para el mejoramiento progresivo, en el caso del sistema a nivel de interfaz del cliente se utilizó el framework de JavaScript The Dojo Toolkit version 1.7 (http://dojotoolkit.org).

IMPLEMENTACIÓN E INSTALACIÓN DEL SISTEMA

El sistema se divide en **tres** carpetas que cuentan con todo el código fuente para operar:

1. Contenido (Carpeta que contiene el motor de todo el sistema)
2. La carpeta **admin** (Place Holder de el motor del sistema)
3. **cursos** (Carpeta que contiene la oferta de cursos del sistema y su pago mediante WebPay)

La carpeta **"contenido"** debe de ser colocada estrictamente en un nivel arriba del Document Root o carpeta pública de Apache, esto se desarrollo así por cuestiones de seguridad apegándose al manual de seguridad para desarrollos web de la guía OWASP (The Open Web Application Security Project) BlackHat Edition 2.0

La carpeta **"admin"** debe de ser colocada en el document Root del apache a nivel público para que pueda integrar de forma correcta el motor del sistema que se encuentra en la carpeta de contenido.

La carpeta de **"cursos"** esta puede residir en el mismo servidor o en otro servidor, en esta se encuentra el sistema de pagos electrónicos de los cursos dados de alta en el sistema.

La instalación del Moodle 1.9.x o 2.2.x se puede realizar en cualquier servidor siempre y cuando el sistema tenga acceso a la base de datos del Moodle sin importar en que servidor resida.

El sistema solo interactúa con la base de datos de Moodle configurada sin afectar ningún archivo o estructura del Moodle, ya que este maneja parámetros y contextos en menoría sin hacer posible la unificación de scripts o programas

CONFIGURACIÓN DEL SISTEMA

El sistema en términos generales tiene cuatro archivos de configuración

1. Archivo de configuración principal
2. Archivo de configuración de conexiones con el moodle version 1.9.x
3. Archivo de configuración de conexiones con el moodle versión 2.2.x
4. Contextos de moodle

En los tres archivos se configura el sistema mediante constantes de PHP si interacción con bases de datos, tanto del sistema, como de Moodle, así como los servicios que utiliza para interactuar con los dos proveedores externos que son Tralix (Facturación electrónica) y MITEC (Pago en línea de cursos)

ARCHIVO DE CONFIGURACION PRINCIPAL:

Ubicación: ***"contenido/conf/conf.php"***

En este archivo se configuran los siguientes parámetros:

1. Conexión con la base de datos MySQL del sistema principal
2. Configuración de la interacción del sistema con el Moodle
3. Configuración de factura electrónica con el proveedor Tralix
4. Configuración de pagos en línea mediate el provedor MITEC (WebPay)
5. Configuración de la ruta para subir comprobantes de pago mediante el sistema

```
<?php
// Archivo general de configuración del sistema
//
//********** CONFIGURACION DE LA CONEXION CON LA BASE DE DATOS MYSQL
define('HOST_DATA_BASE', 'localhost'); // Servidor que tiene corriendo la base de datos.
//define('HOST_DATA_BASE', '10.0.33.51'); // Servidor que tiene corriendo la base de datos.
//define('HOST_DATA_BASE', 'mysql01.sined.svc.vb-anuies.mx'); // Servidor que tiene corriendo la base de datos.
define('DATA_BASE_NAME', 'ANUIESVB'); // Nombre de la Base de Datos de Empresas.
define('LOGIN_ACCESO_DATA_BASE', 'anuiescente'); // Login de Acceso a la Base de Datos.
define('PASS_ACCESO_DATA_BASE', 'Ws.777.AdQe'); // Password de Acceso a la Base de Datos.
define('LLAVE', '!@#$%^&*()_+=-{}][;";/?<>.,'); // Semilla para Tokens y Passwords

// ********** CONFIGURACION CON LA INTERACCION DE MOODLE Y EL SISTEMA
define('MOODLE_ACTIVE', 'v1.9'); // Bandera para indicar la version activa de moodle. Valores permitidos: v1.9, v2.2 y
BOTH
define('MOODLE_V1_9', 'v1.9'); // Constante para las versiones de Moodle v1.9
define('MOODLE_V2_2', 'v2.2'); // Constante para las versiones de Moodle v2.2
define('MOODLE_BOTH', 'BOTH'); // Constante para usar ambas versiones de Moodle al mismo tiempo
define('LOGGER_FILE', '../../log/moodle.log'); //Constante que se usa para llevar una bitacora de movimientos del
sistema
define('LOGGER_LEVEL', 1); // Nivel de bitacora del sistema 0 es para producción, 1 es para desarrollo

// ********** Configuración de factura electrónica con TRALIX
define('CFD_RUTA', '../../facturas');  // Ruta interna donde se guardará el archivo TXT que requiere TRALIX
define('CFD_PREFIJO_ARCHIVO', 'Factura'); //Prefijo del archivo TXT que se utilizara para guardar un historial y envio
de formatos
define('CFD_FTP_USER', 'VALLE-PR'); //Usuario asignado a la ANUIES al servidor de Tralix
define('CFD_FTP_PASSWORD', 'anuies'); // Contraseña de asignada a la ANUIES al servidor de Tralix
define('CFD_FTP_IP', '173.205.255.26'); // Dirección de IP o URL del servidor de Tralix
```

```
define('CFD_FTP_PORT', '22'); // Puerto de conexion al servidor
define('CFD_FTP_TYPE', 'SECURE');//SECURE o UNSECURE // Parametros para definir el tipo de protocolo, ya sea SFTP o FTP según sea el caso

// ********** CONFIGURACIÓN DE PAGOS EN LINEA MEDIANTE WEBPAY
define('ID_ANUIES', 'B002'); //Id de la ANUIES asignado por MITEC
define('SEMILLA_WP', 'MEX001B002'); // Semilla de encripción de la ANUIES asignada por MITEC
define('XMLM',
'5647586E6A376D4C4A384452524F43744E42415236372F466C53585A435077493055546772545151456575746234457837687258A3774694869365A6E6A6C716D56556B6F44362F7848634D410A363562674B5079414C7643776F2F6C714C314A6F70326F36784E66644D2F6B44732B324168366A4B76486D3567576E657971414554306A6B467378703071676C757154324D6E334F746F4D610A454E61416A6C76487369674D455746775765375346F70553642506D484A50766D4A2F3452396D646F6C2B464A5A62426D57593D'); // Cadena XML de encripción asignada por MITEC
define('URL_PAGO', 'https://dev.mitec.com.mx/pgs/WebPay?'); // URL de ventana de pagos asignada por MITEC
define('URL_REPUESTA', 'http://lms.vb-anuies.mx/cursos/confirmacion.php'); //Servidor de respuesta de pagos por MITEC

// ********** Ruta de comprobantes de pagos de los alumnos cuando la operaciíon sea manual
define('RUTA_COMPROBANTE', '../../contenido/pagos/'); // Ruta de los comprobantes de pago donde subira el sistema todos los recibos digitalizados

?>
```

ARCHIVO DE CONFIGURACION PARA MOODLE VERSION 1.9.X:

Ubicación: ***"libs/moodle/v1_9/moodle_config19.php"***

En este archivo se configura la conexión a la base de datos del Moodle versión 1.9.x y la semilla de encripción del moodle, esta debe de ser la misma que usa el Moodle en el archivo config:

```
<?php
// CONFIGURACION DE PARAMETROS DE CONEXION A BASE DE DATOS DEL MOODLE19
$CFG_v19 = new stdClass(); //Creación del objeto de conexion a base de datos con moodle
$CFG_v19->dbtype    = 'mysql'; // Tipo de base de datos que usa moodle
$CFG_v19->dbhost    = 'localhost'; // Servidor de la base de datos de Moodle
//$CFG_v19->dbhost    = 'mysql01.sined.svc.vb-anuies.mx'; // Servidor de la base de datos de Moodle
$CFG_v19->dbname    = 'moodle19'; //Nombre de la base de datos del moodle
$CFG_v19->dbuser    = 'moodle19'; // Nombre del usuario dela base de datos
$CFG_v19->dbpass    = '#UdN23aN'; //Contraseña de conexión con la base de datos de Moodle
$CFG_v19->prefix    = 'mdl_';     // Prefijos de la base de datos de Moodle
$CFG_v19->passwordsaltmain = 'P^#3k.5DfkEd>*5zRQZ[2I&Wp'; // Salt o semilla de encripcion de contraseñas usado en el archivo de configuración del Mooodle

/**
 * Function to connect to the moodle database v1.9
 * @return unknown
 */
function Sdx_ConectaBasev19(){
        global $CFG_v19;
        $dbh = mysqli_connect($CFG_v19->dbhost, $CFG_v19->dbuser, $CFG_v19->dbpass, $CFG_v19->dbname);
//      mysqli_autocommit($dbh,false);
        return $dbh;
}

/**
 * Calculate hashed value from password using current hash mechanism.
```

```
 *
 * @param string password
 * @return string password hash
 */
function hash_internal_user_passwordv19($password) {
        global $CFG_v19;

        if (isset($CFG_v19->passwordsaltmain)) {
                return md5($password.$CFG_v19->passwordsaltmain);
        } else {
                return md5($password);
        }
}
class Configv19{}
?>
```

ARCHIVO DE CONFIGURACION PARA MOODLE VERSION 2.2.X:

Ubicación: ***"libs/moodle/v2_2/moodle_config22.php"***

En este archivo se configura la conexión a la base de datos del Moodle versión 2.2.x y la semilla de encripción del Moodle, esta debe de ser la misma que usa el Moodle en el archivo config:

```
<?php
// CONFIGURACION DE PARAMETROS DE CONEXION A BASE DE DATOS DEL MOODLE VERSIONES 2.2.X
$CFG_v22 = new stdClass(); //Creación del objeto de conexion a base de datos con moodle
$CFG_v22->dbtype    = 'mysql'; // Tipo de base de datos que usa moodle
$CFG_v22->dbhost    = 'localhost'; // Servidor de la base de datos de Moodle
$CFG_v22->dbname    = 'moodle_22'; //Nombre de la base de datos del moodle
$CFG_v22->dbuser    = 'moodle_22'; // Nombre del usuario de la base de datos
$CFG_v22->dbpass    = 'moodle';    //Contraseña de conexión con la base de datos de Moodle
$CFG_v22->prefix    = 'mdl_';       // Prefijos de la base de datos de Moodle
$CFG_v22->passwordsaltmain = 'Y`@#%T*M0W-}{mDk>1[^AzG&Rk>';  // Salt o semilla de encripcion de contraseñas
usado en el archivo de configuración del Mooodle

/**
 * Function to connect to the moodle database v1.9
 * @return unknown
 */
function Sdx_ConectaBasev22(){
        global $CFG_v22;
        $dbh = mysqli_connect($CFG_v22->dbhost, $CFG_v22->dbuser, $CFG_v22->dbpass, $CFG_v22->dbname);
        //mysqli_autocommit($dbh,false);
        return $dbh;
}

/**
 * Calculate hashed value from password using current hash mechanism.
 *
 * @param string password
 * @return string password hash
 */
function hash_internal_user_passwordv22($password) {
        global $CFG_v22;
```

```
        if (isset($CFG_v22->passwordsaltmain)) {
                return md5($password.$CFG_v22->passwordsaltmain);
        } else {
                return md5($password);
        }
}
class Configv22{

}
?>
```

ARCHIVO DE CONFIGURACION DE CONSTANTES DEL MOODLE:

Ubicación: ***"libs/moodle/moodle_constants.php"***

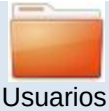
En este archivo se configura los valores de los contextos usados por el Moodle donde afortunadamente en las versiones 1 y 2 ha manejado las mismas constantes, estos contextos los usa para diferenciar los objetos con los que interactua internamente:

```
<?php
// ******** ARCHIVOS DE CONFIGURACION DE CONTEXTOS DEL MOODLE V1.9 Y V2.2
// ******** ESTE PUEDE CAMBIAR EN VERSIONES FUTURAS
define('CONTEXT_USER', 30); // VALOR DEL CONTEXTO DE UN USUARIO EN MOODLE
define('CONTEXT_COURSECAT', 40); // VALOR DEL CONTEXTO DE LAS CATEGORIAS DEL MOODLE
define('CONTEXT_COURSE', 50); // VALOR DEL CONTEXTO DE LOS CURSOS DE MOODLE
define('CONTEXT_BLOCK', 80);  // VALOR DEL CONTEXTO DE BLOQUEO DE OBJETOS EN EL MOODLE

define('TEACHER_ROLE','3');  //  VALOR DEL CONTEXTO DE ROL DE MAESTRO EN MOODLE
define('STUDENT_ROLE','5');  //  VALOR DEL CONTEXTO DE ROL DE ESTUDIANTE EN MOODLE

class MoodleConstants{}
?>
```

ESTRUCTURA DEL SISTEMA

El sistema se divide en dos carpetas principales una es la carpeta pública donde los usuarios podrán consultar el sistema de administración a través de Place Holders, y la otra carpeta es la del contenido que contiene el motor de todo el sistema.

Carpeta	Descripción
contenido	En esta carpeta se encuentran todos los programas de los módulos con que cuenta el sistema, aquí se encuentra el motor principal. Esta carpeta deberá de estar en un nivel arriba de “Document Root” o Raíz de archivos. Y debe de contar con permisos UNIX de 775.
admin	En esta carpeta se encuentran contenidas las carpetas de cada uno de los módulos que contiene el sistema y los programas principales para que funcione correctamente.
Alumnos	En esta carpeta se encuentran los scripts de PHP y programas con la información de alumnos para su registro, modificación y eliminación del sistema y además de contener los scripts de sincronización con el Moodle.
AlumnosGrupo	En esta carpeta se encuentran los scripts de PHP y programas con la información de alumnos por grupo y los programas que dan de alta en el Moodle.
CatCursos	En esta carpeta se encuentran los scripts de PHP y programas con la información de las categorías de cursos disponibles y su alta y modificación con el Moodle
Constancias	En esta carpeta se encuentran los scripts de PHP y programas para la generación de Constancias de alumnos por curso.
Cursos	En esta carpeta se encuentran los scripts de PHP y programas con la información de Cursos, además de su alta y modificación con el Moodle.
Empresas	En esta carpeta se encuentran los scripts de PHP y programas con la información del Catálogo de Empresas e Instituciones.
Equipo	En esta carpeta se encuentran los scripts de PHP y programas con la información del Catálogos de Equipo.
Espacios	En esta carpeta se encuentran los scripts de PHP y programas con la información del Catálogo de Espacios.
Eventos	En esta carpeta se encuentran los scripts de PHP y programas con la información de la Administración de Eventos.
	En esta carpeta se encuentran los scripts de PHP y programas con la información de los Catálogo de Giros Comerciales.

Giros	
Habitaciones	En esta carpeta se encuentran los scripts de PHP y programas con la información del Catálogo de Habitaciones.
Inicio	En esta carpeta se encuentran los scripts de PHP y programas con la información de Inicio.
Instructores	En esta carpeta se encuentran los scripts de PHP y programas para el módulo que manipula la información de los Instructores dentro del sistema y en Moodle.
OHospedaje	En esta carpeta se encuentran los scripts de PHP y programas de Catálogo de Tipos de Ocupación de Habitaciones.
Reporte Académico	En esta carpeta se encuentran los scripts de PHP y programas que generan de los reportes académicos
SAlimentos	En esta carpeta se encuentran los scripts de PHP y programas para los Catálogos de Servicios de Alimentos
SVarios	En esta carpeta se encuentran los scripts de PHP y programas con los Catálogos de Servicios Varios.
TEventos	En esta carpeta se encuentran los scripts de PHP y programas que dan de alta, modifican o eliminan información de Eventos que están relacionados con la impartición de algún curso, coloquio, asamblea, etc.
TSAlimentos	En esta carpeta se encuentran los scripts de PHP y programas que dan de alta, modifican o eliminan información de los Catálogos de Tipos de Servicios de Alimentos.
TSectores	En esta carpeta se encuentran los scripts de PHP y programas que dan de alta, modifican o eliminan información de los Catálogos de Sectores de Procedencia.
Ubicaciones	En esta carpeta se encuentran los scripts de PHP y programas con la información de los Catálogos de Ubicación de Espacios.
UHospedaje	En esta carpeta se encuentran los scripts de PHP y programas con los Catálogos de Ubicaciones Hospedaje
 Usuarios	En esta carpeta se encuentran los scripts de PHP y programas que dan de alta, modifican, eliminan a información del usuario.
conf	En esta carpeta se encuentran los scripts y programas de la configuración del sistema.

libs	En esta carpeta se encuentran las Librerías en Lenguaje PHP utilizadas para el sistema. (Esta carpeta deberá de estar en un subnivel de la carpeta de contenido).
KLogger	En esta carpeta se encuentra la Librería para el uso del Klogger.
moodle	En esta carpeta se encuentra el sistema los scripts para el funcionamiento de Moodle.
pagos	En esta carpeta se encuentra los comprobantes de los estudiantes que pagaron con un depósito bancario.
phpexcel	En esta carpeta se encuentra la Librería PhpExcel que permite generar documentos de Excel desde el sistema usando solo PHP.
phpseclib	En esta carpeta se encuentra la Librería por secciones.
reportes	En esta carpeta se encuentra se almacenados los reportes generados por el sistema como PDF, Excel.
tcpdf	En esta carpeta se encuentra la Librería TcPDF que permite generar documentos en formato PDF desde el sistema usando solo PHP.
trailix	En esta carpeta se encuentra la Librería para generar facturas electrónicas con el proveedor Trailix.
util	En esta carpeta se encuentra la Librería de utilidades de texto.

Carpeta	Descripción
public	Esta carpeta simboliza el Document Root del sistema y todos los programas necesarios para su despliegue y operación.
admin	Esta carpeta contiene los Scripts que mandan llamar los programas necesarios para visualizar el Sistema, estos programas ingresan directamente al sistema que se encuentra en la carpeta de Contenido (dos niveles arriba).
rsrcs	Se encuentran los Recursos usados por el portal que son Imágenes complementarias al diseño del portal como son: Texturas, Botones, Cascadas de Estilo, y Scripts de JavaScript.
fonts	Esta carpeta contiene las Imágenes que el Sistema requiere para su operación y correcto despliegue de reportes en PDF.
imgs	Se encuentran las Imágenes Básicas que arman la estructura de diseño del portal.
js	Se encuentran los Scripts de JavaScript que le dan funcionalidad al portal.
dijit	Contiene las librerías Dijit del Framework Dojo.
dojo	Contiene el Motor Principal del Framework Dojo.
dojox	Contiene las Librerías del Dojox o Widgets creados por Sandox Systems para Dojo.
sandox	En esta carpeta se encuentran Librerías en JavaScript, creadas por Sandox Systems para funciones adicionales del sistema.
util	En esta carpeta se puede encontrar una versión compilada del Dojo de todas las Librerías utilizadas por Sandox Systems para el funcionamiento del sistema.
styles	Se encuentran las Cascadas de Estilo (CSS) para el funcionamiento del portal.
cursos	En esta carpeta se encuentran la oferta de cursos para quienes están interesados en inscribirse a uno, u obtener más información acercad e estos o pagar un curso vía web
recursos	Esta carpeta contiene los Scripts que mandan llamar los programas necesarios para visualizar el sistema, estos programas interactúan directamente al Sistema que se encuentra en la carpeta de Contenido (dos niveles arriba).

estilos	Esta carpeta contiene todos los recursos que ocupa el sistema que son: Imágenes, Scripts de JavaScript, y Cascadas de Estilo (CSS).
imagenes	Se encuentran las Imágenes que usa el sistema.
resources	Esta carpeta contiene los Recursos del sistema.
fonts	Esta carpeta contiene los tipos de fuentes soportadas por el sistema.
tools	Esta carpeta contiene los diccionarios que ocupa el sistema
words	Esta carpeta contiene un diccionario de palabras para generar una imagen tipo captcha.

FUNCIONAMIENTO DEL SISTEMA

El sistema consiste de una página Autenticación de Usuarios y los Módulos que se explican a continuación:

Generalidades de los módulos del sistema

El sistema se opera a través de 4 archivos básicos que se explican a continuación:

Armador.php	Este programa arma Contenidos de Interfaces únicamente, ya sea de HTML y JavaScript en conjunto, verificando antes el Nivel de Acceso del Usuario y la Sesión, además de contener Directivas de Seguridad.
FUP.php	Este programa se encarga de Verificar el Progreso de Subida de Archivos con ayuda del módulo APC instalado en PHP.
RPC.php	Este programa se encarga de atender Llamadas de Procedimiento Remoto que requieren de comunicación con la base de datos.

Este programa atiende Llamadas de Procedimiento Remoto y se utiliza únicamente para subir información con archivos al servidor.

Todos los módulos contenidos en la carpeta “contenido/admin/Modulo”, (donde “Modulo” es el nombre del módulo), son inicializados por el programa “Armador.php” que se puede encontrar en la carpeta “contenido/admin” llamando al archivo Intro.php que construye la página principal de cada módulo.

Todos los archivos que empiezan con el prefijo “rpc” representan Procedimientos Remotos, obtienen sus datos por el método POST y generalmente realizan acciones sobre la base de datos. El resultado de su ejecución genera JSON (http://www.json.org).

Existen estos principalmente:

- Los que empiezan con “rpcRegistra”: Dan de alta nuevos registros en la base de datos.
- Los que empiezan con “rpcModifica”: Actualiza registros existentes en base de datos.
- Los que empiezan con “rpcElimina”: Elimina registros existentes en base de datos.
- Los que empiezan con “rpcConsulta”: Realizan una consulta sobre la base de datos.
- Los que empiezan con “rpcBusca”: Obtiene todos los campos de un registro en la base de datos.

  Los archivos que no empiezan con el prefijo “rpc”, reciben información de Pantallas y obtienen sus datos a través del método GET. Nunca realizan cambios en la base de datos. Como resultado de la ejecución generan HTML.

Existen estos principalmente:

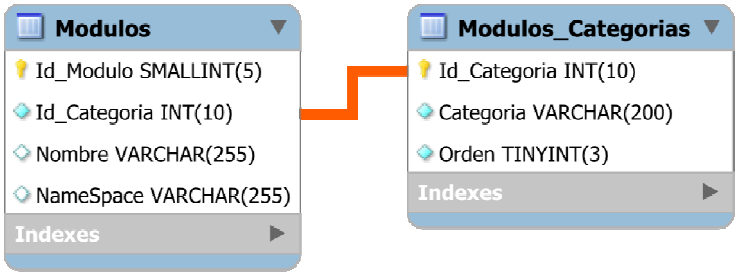
- Los que se llaman Intro.php: Generan el Formulario para dar de alta, modificar consultar y eliminar registros.

La tabla de la base de datos que liga los NameSpaces de los Módulos es la tabla “Modulos”.

DICCIONARIO DE DATOS POR MODULO

La base de datos del sistema esta construida con el engine de MyISAM de MySQL y las tablas representadas por sus diagramas las enumeramos a continuación por modulo.

Tabla utilizada por el módulo Módulos



Nombre de la Tabla: "módulos"

En esta tabla se almacena la meta información de los módulos.

Column	Type	Null	Default	Comments
Id_Modulo	smallint(5)	No		Llave primaria
Id_Categoria	int(10)	No		Indice hacia la tabla Modulos_Categorias
Nombre	varchar(255)	Yes	NULL	Nombre completo del módulo
NameSpace	varchar(255)	Yes	NULL	Vincula el módulo a una carpeta en el servidor

Índices de la Tabla: "módulos"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Modulo	23	A		

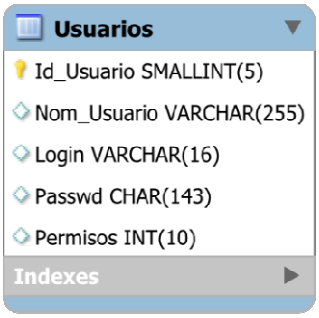
Nombre de la Tabla: Modulos_Categorias

Column	Type	Null	Default	Comments
Id_Categoria	int(10)	No		Llave Primaria de la Tabla Modulos_Categorias
Categoria	varchar(200)	No		Categoria del modulo
Orden	tinyint(3)	No		Orden de categoria

Índice de la Tabla: Modulos_Categorias

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Categoria	5	A		

Tabla utilizada por el módulo Usuarios



Nombre de la Tabla: "Usuarios"

En esta tabla se almacenan los Usuarios del Sistema.

Column	Type	Null	Default	Comments
Id_Usuario	smallint(5)	No		Llave primaria
Nom_Usuario	varchar(255)	Yes	NULL	Nombre completo del usuario
Login	varchar(16)	Yes	NULL	Login del usuario
Passwd	char(143)	Yes	NULL	Password encriptado del usuario incluyendo la salt
Permisos	int(10)	Yes	NULL	

Índices de la Tabla: " Usuarios"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Usuario	9	A		

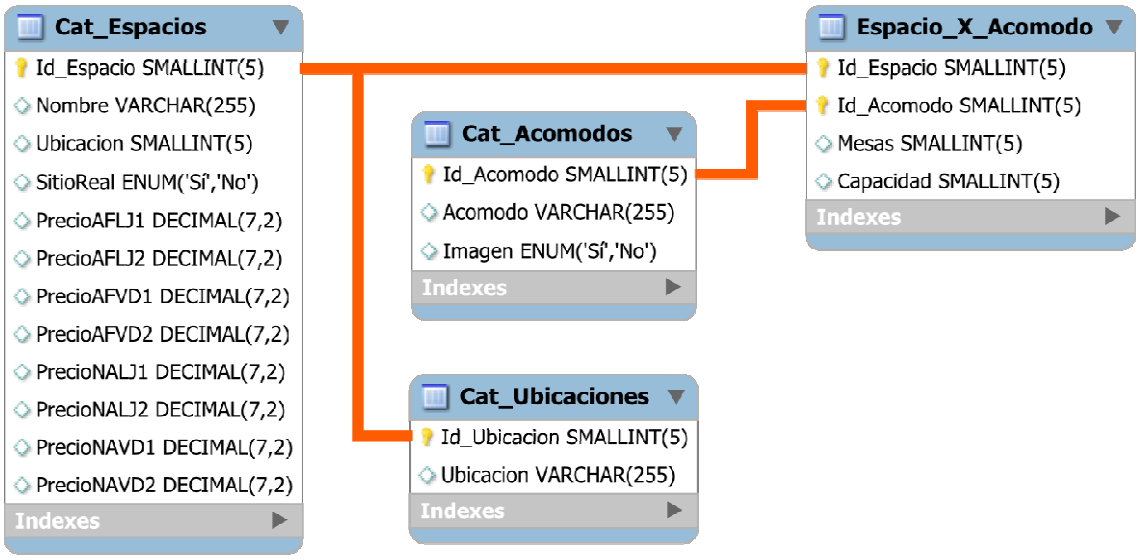
Nombre de la Tabla: Sessions

Column	Type	Null	Default	Comments
session_id	varchar(100)	No		Id de la session
session_data	text	No		Datos de la session
expires	int(11)	No	0	Tiempo en session

Índice de la Tabla: Sessions

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	session_id	139	A		

Tablas Utilizadas por los módulos Espacios



Nombre de la Tabla: "Cat_Ubicaciones".

Column	Type	Null	Default	Comments
Id_Ubicacion	smallint(5)	No		Llave Primaria de la Tabla Cat_Ubicaciones
Ubicacion	varchar(255)	Yes	NULL	Ubicación de las Habitaciones

Índice de la Tabla: "Cat_Ubicaciones"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Ubicacion	2	A		

Nombre de la Tabla: "Cat_Acomodos"

Column	Type	Null	Default	Comments
Id_Acomodo	smallint(5)	No		Llave Primaria de la Tabla Cat_Acomodos
Acomodo	varchar(255)	Yes	NULL	Lugares de Acomodo
Imagen	enum('Sí', 'No')	Yes	NULL	Si cuenta o No con imagen para mostrar

Índice de la Tabla: "Cat_Acomodos"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Acomodo	6	A		

Nombre de la Tabla: "Cat_Espacios"

Column	Type	Null	Default	Comments
Id_Espacio	smallint(5)	No		Llave Primaria de la Tabla Cat_Espacios
Nombre	varchar(255)	Yes	NULL	Nombre del espacio donde sera utilizado algun equipo
Ubicacion	smallint(5)	Yes	NULL	Lugar de ubicación
SitioReal	enum('Sí', 'No')	Yes	NULL	Sitio Real Fisicamente
PrecioAFLJ1	decimal(7,2)	Yes	NULL	Precio afiliadas Lunes a jueves 1
PrecioAFLJ2	decimal(7,2)	Yes	NULL	Precio afiliadas Lunes a jueves 2
PrecioAFVD1	decimal(7,2)	Yes	NULL	Precio afiliadas viernes a domingo 1
PrecioAFVD2	decimal(7,2)	Yes	NULL	Precio afiliadas viernes a domingo 2
PrecioNALJ1	decimal(7,2)	Yes	NULL	Precio no afiliadas lunes a jueves 1
PrecioNALJ2	decimal(7,2)	Yes	NULL	Precio no afiliadas lunes a jueves 2
PrecioNAVD1	decimal(7,2)	Yes	NULL	Precio no afiliadas viernes a domingo 1
PrecioNAVD2	decimal(7,2)	Yes	NULL	Precio no afiliadas viernes a domingo 2

Índice de la Tabla: "Cat_Espacios"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Espacio	13	A		

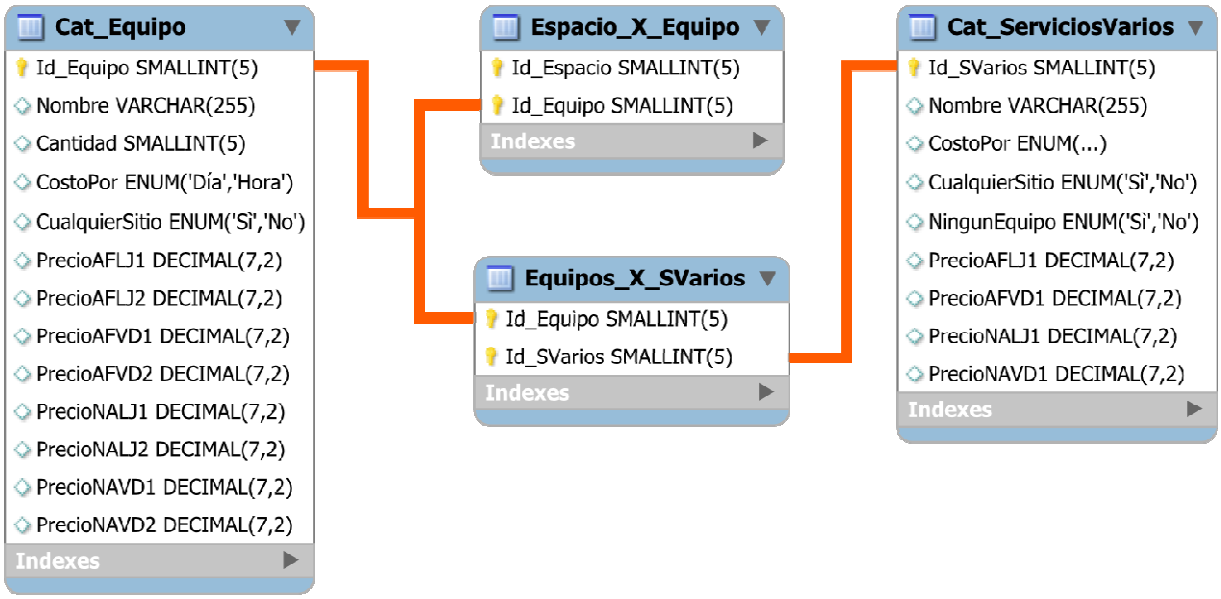
Nombre de la Tabla: "Espacio_X_Acomodo"

Column	Type	Null	Default	Comments
Id_Espacio	smallint(5)	No		Indice hacia la tabla Cat_Espacios
Id_Acomodo	smallint(5)	No		Indice hacia la tabla Cat_Acomodos
Mesas	smallint(5)	Yes	NULL	Numero de mesas a utilizar
Capacidad	smallint(5)	Yes	NULL	Capacidad total

Índice de la Tabla: "Espacio_X_Acomodo"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Espacio	0	A		
				Id_Acomodo	37	A		

Tablas Utilizadas por los módulos Equipos y Servicios Varios



Nombre de la Tabla: "Cat_Equipo"

Column	Type	Null	Default	Comments
Id_Equipo	smallint(5)	No		Llave Primaria de la Tabla Cat_Equipo
Nombre	varchar(255)	Yes	NULL	Nombre del equipo que se utilizará
Cantidad	smallint(5)	Yes	NULL	Cantidad de Equipos a utilizar
CostoPor	enum('Día', 'Hora')	Yes	NULL	Costo por Dia u Hora, que seran utilizado
CualquierSitio	enum('Si', 'No')	Yes	NULL	Si el equipo se puede utilizar en cialquier sitio
PrecioAFLJ1	decimal(7,2)	Yes	NULL	Precio afiliadas Lunes a jueves 1
PrecioAFLJ2	decimal(7,2)	Yes	NULL	Precio afiliadas Lunes a jueves 2
PrecioAFVD1	decimal(7,2)	Yes	NULL	Precio afiliadas viernes a domingo 1
PrecioAFVD2	decimal(7,2)	Yes	NULL	Precio afiliadas viernes a domingo 2
PrecioNALJ1	decimal(7,2)	Yes	NULL	Precio no afiliadas lunes a jueves 1
PrecioNALJ2	decimal(7,2)	Yes	NULL	Precio no afiliadas lunes a jueves 2
PrecioNAVD1	decimal(7,2)	Yes	NULL	Precio no afiliadas viernes a domingo 1
PrecioNAVD2	decimal(7,2)	Yes	NULL	Precio no afiliadas viernes a domingo 2

Índice de la Tabla: "Cat_Equipo"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Equipo	26	A		

Nombre de la Tabla: "Espacio_X_Equipo"

Column	Type	Null	Default	Comments
Id_Espacio	smallint(5)	No		Indice hacia la tabla Cat_Espacios
Id_Equipo	smallint(5)	No		Indice hacia la tabla Cat_Equipo

Índice de la Tabla: "Espacio_X_Equipo"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Espacio	0	A		
				Id_Equipo	23	A		

Nombre de la Tabla: "Cat_ServiciosVarios"

Column	Type	Null	Default	Comments
Id_SVarios	smallint(5)	No		Llave Primaria de la Tabla Cat_ServiciosVarios
Nombre	varchar(255)	Yes	NULL	Nombre del Servicio
CostoPor	enum('Día', 'Hora', 'Minuto', 'Unidad')	Yes	NULL	Costo por tiempo que seran utilizado
CualquierSitio	enum('Sì', 'No')	Yes	NULL	Posibilidad de usarse en lugares no habilitados
NingunEquipo	enum('Sì', 'No')	Yes	NULL	
PrecioAFLJ1	decimal(7,2)	Yes	NULL	Precio afiliadas Lunes a jueves 1
PrecioAFVD1	decimal(7,2)	Yes	NULL	Precio afiliadas viernes a domingo 1
PrecioNALJ1	decimal(7,2)	Yes	NULL	Precio no afiliadas lunes a jueves 1
PrecioNAVD1	decimal(7,2)	Yes	NULL	Precio no afiliadas viernes a domingo 1

Índice de la Tabla: "Cat_ServiciosVarios"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_SVarios	20	A		

Nombre de la Tabla: "Espacio_X_SVarios"

Column	Type	Null	Default	Comments
Id_Espacio	smallint(5)	No		Indice hacia la tabla Cat_Espacios
Id_SVarios	smallint(5)	No		Indice hacia la tabla Cat_ServiciosVarios

Índice de la Tabla: Espacio_X_SVarios

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Espacio	0	A		
				Id_SVarios	13	A		

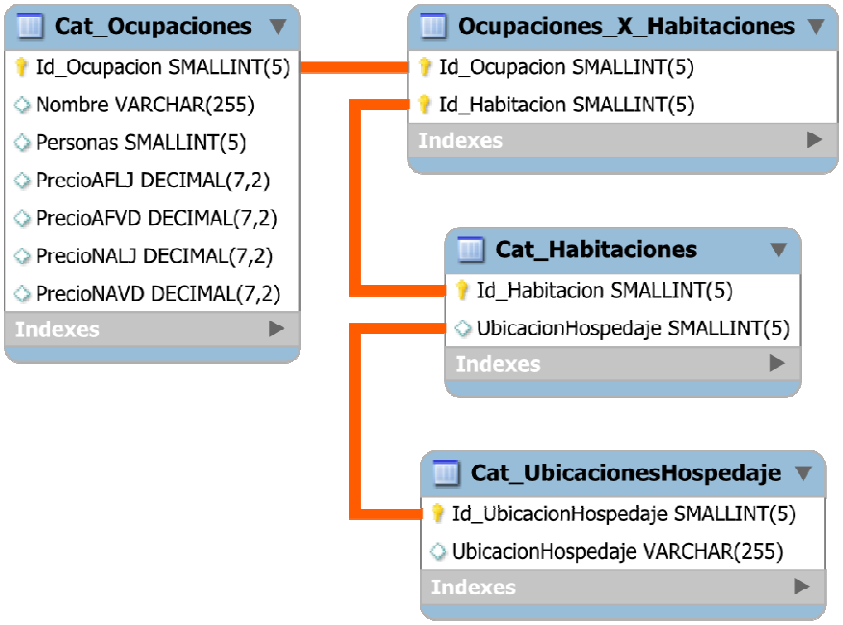
Nombre de la Tabla: "Equipos_X_SVarios"

Column	Type	Null	Default	Comments
Id_Equipo	smallint(5)	No		Indice hacia la tabla Cat_Equipo
Id_SVarios	smallint(5)	No		Indice hacia la tabla Cat_ServiciosVarios

Índice de la Tabla: Equipos_X_SVarios

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Equipo	0	A		
				Id_SVarios	3	A		

Tablas Utilizadas por los módulos Hospedaje



Nombre de la Tabla: "Cat_UbicacionesHospedaje"

Column	Type	Null	Default	Comments
Id_UbicacionHospedaje	smallint(5)	No		Llave Primaria de la Tabla Cat_UbicacionHospedaje
UbicacionHospedaje	varchar(255)	Yes	NULL	Ubicación de Hospedaje

Índice de la Tabla: "Cat_UbicacionesHospedaje"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_UbicacionHospedaje	3	A		

Nombre de la Tabla: "Cat_Ocupaciones".

Column	Type	Null	Default	Comments
Id_Ocupacion	smallint(5)	No		Llave Primaria de la Tabla Cat_ocupaciones
Nombre	varchar(255)	Yes	NULL	Tipo de la Habitación
Personas	smallint(5)	Yes	NULL	Numero de personas en la habitación
PrecioAFLJ	decimal(7,2)	Yes	NULL	Precio afiliadas lunes a jueves
PrecioAFVD	decimal(7,2)	Yes	NULL	Precio afiliadas viernes a domingo
PrecioNALJ	decimal(7,2)	Yes	NULL	Precio no afiliadas lunes a jueves
PrecioNAVD	decimal(7,2)	Yes	NULL	Precio no afiliadas viernes a jueves

Índice de la Tabla: "Cat_Ocupaciones"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Ocupacion	5	A		

Nombre de la Tabla: "Cat_Habitaciones"

Column	Type	Null	Default	Comments
Id_Habitacion	smallint(5)	No		Llave Primaria de la Tabla Cat_Habitaciones
UbicacionHospedaje	smallint(5)	Yes	NULL	Piso en el que se encuentra su lugar de hospedaje

Índice de la Tabla: "Cat_Habitaciones"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Habitacion	19	A		

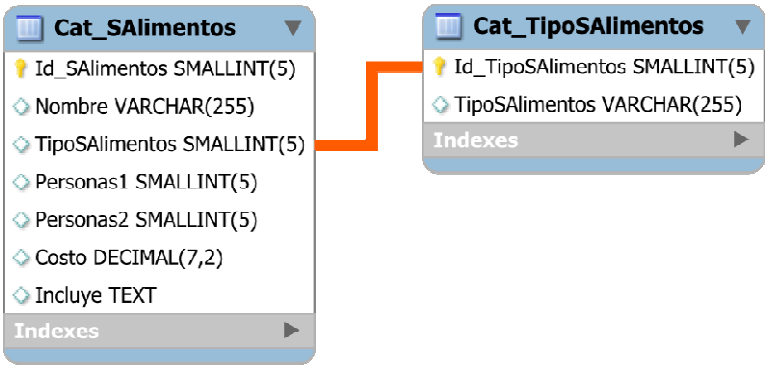
Nombre de la Tabla: "Ocupaciones_X_Habitaciones".

Column	Type	Null	Default	Comments
Id_Ocupacion	smallint(5)	No		Indice hacia la tabla Cat_Ocupaciones
Id_Habitacion	smallint(5)	No		Indice hacia la tabla Cat_Habitaciones

Índice de la Tabla: "Ocupaciones_X_Habitaciones"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Ocupacion	0	A		
				Id_Habitacion	38	A		

Tablas Utilizadas por los módulos Servicios de Alimentos



Nombre de la Tabla: "Cat_TiposSAlimentos"

Column	Type	Null	Default	Comments
Id_TipoSAlimentos	smallint(5)	No		Llave Primaria de la Tabla Cat_TiposSAlimentos
TipoSAlimentos	varchar(255)	Yes	NULL	Tipo de Alimentos

Índice de la Tabla: "Cat_TiposSAlimentos"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_TipoSAlimentos	3	A		

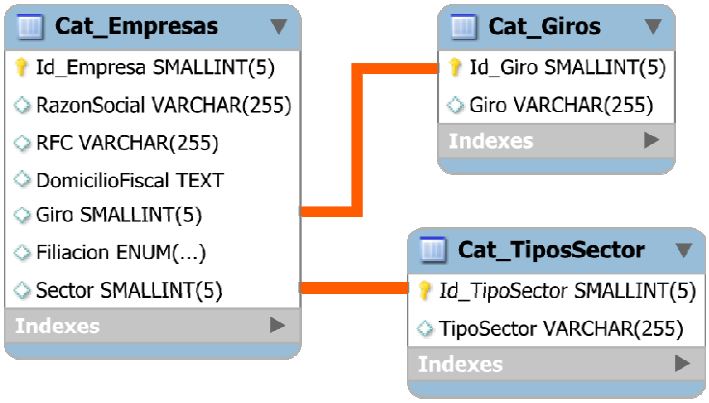
Nombre de la Tabla: "Cat_SAlimentos"

Column	Type	Null	Default	Comments
Id_SAlimentos	smallint(5)	No		Llave Primaria de la Tabla Cat_SAlimentos
Nombre	varchar(255)	Yes	NULL	Nombre del Platillo
TipoSAlimentos	smallint(5)	Yes	NULL	Tipo de Alimentos
Personas1	smallint(5)	Yes	NULL	Numero de personas 1
Personas2	smallint(5)	Yes	NULL	Numero de personas 2
Costo	decimal(7,2)	Yes	NULL	Costo del Platillo
Incluye	text	Yes	NULL	Descripción de lo que incluye el platillo

Índice de la Tabla: "Cat_SAlimentos"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_SAlimentos	21	A		

Tablas Utilizadas por los Módulos Empresas e Instituciones



Nombre de la Tabla: "Cat_Giros"

Column	Type	Null	Default	Comments
Id_Giro	smallint(5)	No		Llave Primaria de la Tabla Cat_Giros
Giro	varchar(255)	Yes	NULL	Tipo de Comercio

Índice de la Tabla: "Cat_Giros"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Giro	27	A		

Nombre de la Tabla: "Cat_TiposSector"

Column	Type	Null	Default	Comments
Id_TipoEvento	smallint(5)	No		Llave Primaria de la Tabla Cat_TiposEvento
TipoEvento	varchar(255)	Yes	NULL	Tipo de Evento

Índice de la Tabla: "Cat_TiposSector"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_TipoEvento	14	A		

Nombre de la Tabla: "Cat_Empresas"

Column	Type	Null	Default	Comments
Id_Empresa	smallint(5)	No		Llave Primaria de la Tabla Cat_Empresas
RazonSocial	varchar(255)	Yes	NULL	Razón Social
RFC	varchar(255)	Yes	NULL	Registro Federal de Contribuyentes
DomicilioFiscal	text	Yes	NULL	Domicilio de la ubicación de la Empresa
Giro	smallint(5)	Yes	NULL	Giro de trabajo al que pertenece la Empresa

Filiacion	enum('Afiliada', 'No Afiliada')	Yes	NULL	Estado de la Filiación
Sector	smallint(5)	Yes	NULL	Sector al que pertenece

Índice de la Tabla: "Cat_Empresas"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Empresa	2	A		

Nombre de la Tabla: "Cat_Clientes"

Column	Type	Null	Default	Comments
Id_Empresa	smallint(5)	No		Indice hacia la tabla Cat_Empresas
Id_Cliente	smallint(5)	No		Llave Primaria de la Tabla Cat_Clientes
Nombre	varchar(255)	Yes	NULL	Nombre del Cliente
Puesto	varchar(255)	Yes	NULL	Puesto del Cliente
Telefono	varchar(255)	Yes	NULL	Telefono Particular
Celular	varchar(255)	Yes	NULL	Telefono Celular
Correo	varchar(255)	Yes	NULL	Correo electrónico

Índice de la Tabla: "Cat_Clientes"

Keyname	Type	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	No	Id_Empresa	0	A		
			Id_Cliente	2	A		

Nombre de la Tabla: "Cat_TiposEventos"

Column	Type	Null	Default	Comments
Id_TipoEvento	smallint(5)	No		Llave Primaria de la Tabla Cat_TiposEvento
TipoEvento	varchar(255)	Yes	NULL	Tipo de Evento

Índice de la Tabla: "Cat_TiposEventos"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_TipoEvento	14	A		

Nombre de la Tabla: "Cat_Cursos"

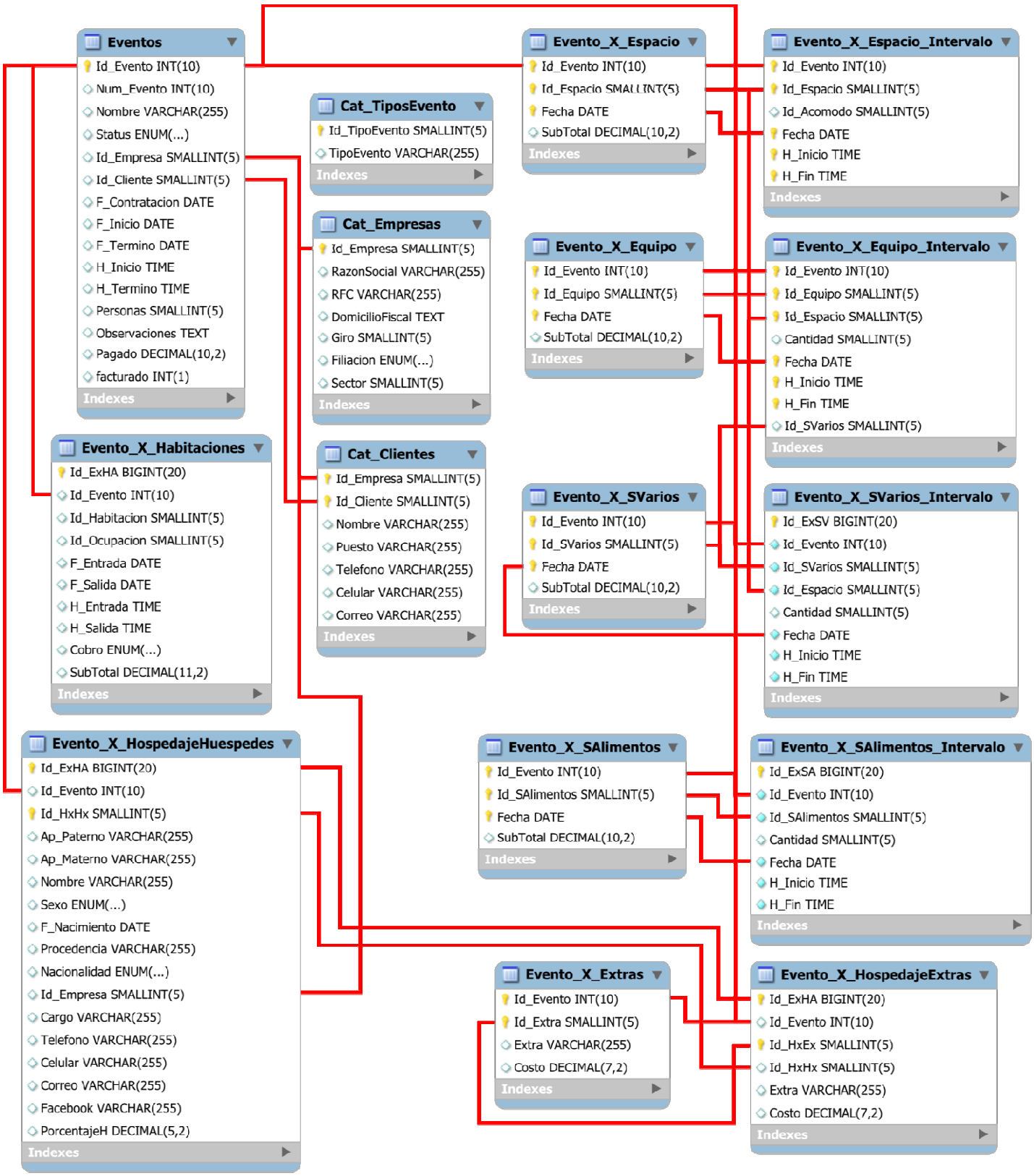
Column	Type	Null	Default	Comments
Id_CatCurso	int(10)	No		Llave Primaria de la Tabla cat_cursos
Nombre	varchar(60)	No		Nombres del curso que se impartirá
Descripcion	text	Yes	NULL	Descripción del curso

Id_Padre	int(10)	Yes	NULL	
id_moodle19	bigint(10)	Yes	NULL	Id del Alumno en el mo0odle version 1.9
id_moodle22	bigint(10)	Yes	NULL	Id del Alumno en el mo0odle version 2.2

Índice de la Tabla:"Cat_Cursos"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_CatCurso	1	A		

Tablas Utilizadas por el módulo Eventos



Nombre de la Tabla: "Eventos"

Column	Type	Null	Default	Comments
Id_Evento	int(10)	No		Llave Primaria de la Tabla Eventos
Num_Evento	int(10)	Yes	NULL	Numero de Evento
Nombre	varchar(255)	Yes	NULL	Nombre del evento
Status	enum('Draft', 'Prospecto', 'Confirmado', 'Cancelado', 'Terminado')	Yes	NULL	Estatus del Evento
Id_Empresa	smallint(5)	Yes	NULL	Indice hacia la Tabla Cat_Empresas
Id_Cliente	smallint(5)	Yes	NULL	Indice hacia la Tabla Cat_Clientes
F_Contratacion	date	Yes	NULL	Fecha de contratación
F_Inicio	date	Yes	NULL	Fecha de inicio
F_Termino	date	Yes	NULL	Fecha de Finalización
H_Inicio	time	Yes	NULL	Hora de inicio
H_Termino	time	Yes	NULL	Hora de Finalización
Personas	smallint(5)	Yes	NULL	Numero de personas
Observaciones	text	Yes	NULL	Observaciones
Pagado	decimal(10,2)	Yes	NULL	Cantidad Pagada
facturado	int(1)	Yes	NULL	Facturación

Índice de la Tabla: "Eventos"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Evento	4	A		

Nombre de la Tabla: "Evento_X_Equipo"

Column	Type	Null	Default	Comments
Id_Evento	int(10)	No		Indice hacia la tabla eventos
Id_Equipo	smallint(5)	No		Indice hacia la tabla Cat_Equipo
Fecha	date	No		Fecha
SubTotal	decimal(10,2)	Yes	NULL	Subtotal de Pago

Índice de la Tabla: "Evento_X_Equipo"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Evento	0	A		
				Id_Equipo	0	A		
				Fecha	14	A		

Nombre de la Tabla: "Evento_X_Equipo_Intervalo"

Column	Type	Null	Default	Comments
Id_Evento	int(10)	No		Indice hacia la tabla Eventos
Id_Equipo	smallint(5)	No		Indice hacia la tabla Cat_Equipo
Id_Espacio	smallint(5)	No	0	Indice hacia la tabla Cat_Espacios
Cantidad	smallint(5)	Yes	NULL	Cantidad de equipo
Fecha	date	No		Fecha de Evento
H_Inicio	time	No		Hora de inicio
H_Fin	time	No		Hora de Finalización
Id_SVarios	smallint(5)	Yes	0	Indice hacia la tabla Cat_EventosVarios

Índice de la Tabla: "Evento_X_Equipo_Intervalo"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Evento	0	A		
				Id_Equipo	0	A		
				Id_Espacio	0	A		
				Fecha	0	A		
				H_Inicio	0	A		
				H_Fin	28	A		

Nombre de la Tabla: "Evento_X_Espacio"

Column	Type	Null	Default	Comments
Id_Evento	int(10)	No		Llave Primaria de la Tabla Eventos
Id_Espacio	smallint(5)	No		Indice hacia la tabla Cat_Espacios
Fecha	date	No		Fecha de evento
SubTotal	decimal(10,2)	Yes	NULL	Subtotal de Pago

Índice de la Tabla: "Evento_X_Espacio"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Evento	0	A		
				Fecha	0	A		
				Id_Espacio	21	A		

Nombre de la Tabla: "Evento_X_Espacio_intervalo"

Column	Type	Null	Default	Comments
Id_Evento	int(10)	No		Indice hacia la tabla Eventos
Id_Espacio	smallint(5)	No		Indice hacia la tabla Cat_Espacios
Id_Acomodo	smallint(5)	Yes	NULL	Indice hacia la Tabla Cat_Acomodos
Fecha	date	No		Fecha de Evento
H_Inicio	time	No		Hora de inicio
H_Fin	time	No		Hora de Finalización

Índice de la Tabla: "Evento_X_Espacio_intervalo"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Evento	0	A		
				Fecha	0	A		
				Id_Espacio	0	A		
				H_Inicio	0	A		
				H_Fin	21	A		

Nombre de la Tabla: "Evento_X_Extras"

Column	Type	Null	Default	Comments
Id_Evento	int(10)	No		Indice hacia la tabla Eventos
Id_Extra	smallint(5)	No		Llave Primaria de la Tabla Evento_X_Extras
Extra	varchar(255)	Yes	NULL	Recursos adicionales
Costo	decimal(7,2)	Yes	NULL	Costo Extra

Índice de la Tabla: "Evento_X_Extras"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Evento	0	A		

Nombre de la Tabla: "Evento_X_Habitaciones"

Column	Type	Null	Default	Comments
Id_ExHA	bigint(20)	No		Llave Primaria de la Tabla Evento_X_Habitaciones
Id_Evento	int(10)	Yes	NULL	Indice hacia la tabla Eventos
Id_Habitacion	smallint(5)	Yes	NULL	Indice hacia la Tabla Cat_Habitaciones
Id_Ocupacion	smallint(5)	Yes	NULL	Indice hacia la Tabla Cat_Ocupaciones
F_Entrada	date	Yes	NULL	Fecha de Entrada
F_Salida	date	Yes	NULL	Fecha de Salida
H_Entrada	time	Yes	NULL	Hora de Entrada
H_Salida	time	Yes	NULL	Hora de Salida
Cobro	enum('Afiliadas', 'No Afiliadas')	Yes	NULL	Cobro dependiendo de la filiación
SubTotal	decimal(11,2)	Yes	NULL	Subtotal a pagar

Índice de la Tabla: "Evento_X_Habitaciones"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_ExHA	0	A		

Nombre de la Tabla: "Evento_X_HospedajeExtras"

Column	Type	Null	Default	Comments
Id_ExHA	bigint(20)	No		Indice hacia la tabla Evento_X_Habitaciones
Id_Evento	int(10)	Yes	NULL	Indice hacia la tabla Eventos
Id_HxEx	smallint(5)	No		Id de la tabla Evento_X_Extra
Id_HxHx	smallint(5)	Yes	NULL	Id del huesped hospedado por habitacion
Extra	varchar(255)	Yes	NULL	Elementos adicionales
Costo	decimal(7,2)	Yes	NULL	Costo extra

Índice de la Tabla:"Evento_X_HospedajeExtras"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_ExHA	0	A		
				Id_HxEx	0	A		

Nombre de la Tabla: "Evento_X_HospedajeHuespedes"

Column	Type	Null	Default	Comments
Id_ExHA	bigint(20)	No		Indice hacia la tabla Evento_X_Habitaciones
Id_Evento	int(10)	Yes	NULL	Indice hacia la tabla Eventos
Id_HxHx	smallint(5)	No		Id de la tabla Evento_X_Hospedaje
Ap_Paterno	varchar(255)	Yes	NULL	Apellido Paterno
Ap_Materno	varchar(255)	Yes	NULL	Apellido Materno
Nombre	varchar(255)	Yes	NULL	Nombres
Sexo	enum('Hombre', 'Mujer')	Yes	NULL	Sexo

F_Nacimiento	date	Yes	NULL	Fecha de nacimiento
Procedencia	varchar(255)	Yes	NULL	Lugar de procedencia
Nacionalidad	enum('Mexicano', 'Extranjero')	Yes	NULL	Nacionalidad
Id_Empresa	smallint(5)	Yes	NULL	Indice hacia la Tabla Cat_Empresas
Cargo	varchar(255)	Yes	NULL	Cargo
Telefono	varchar(255)	Yes	NULL	Telefono Particular
Celular	varchar(255)	Yes	NULL	Telefono Celular
Correo	varchar(255)	Yes	NULL	Correo electrónico
Facebook	varchar(255)	Yes	NULL	Facebook
PorcentajeH	decimal(5,2)	Yes	NULL	Porcentaje de hospedaje

Índice de la Tabla: “Evento_X_HospedajeHuespedes”

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_ExHA	0	A		
				Id_HxHx	0	A		

Nombre de la Tabla: “Evento_X_SAlimentos”

Column	Type	Null	Default	Comments
Id_Evento	int(10)	No		Indice hacia la tabla Eventos
Id_SAlimentos	smallint(5)	No		Indice hacia la tabla Cat_SAlimentos
Fecha	date	No		Fecha
SubTotal	decimal(10,2)	Yes	NULL	Subtotal de Pago

Índice de la Tabla: “Evento_X_SAlimentos”

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Evento	0	A		
				Fecha	0	A		
				Id_SAlimentos	10	A		

Nombre de la Tabla: “Evento_X_SAlimentos_Intervalo”

Column	Type	Null	Default	Comments
Id_ExSA	bigint(20)	No		Llave Primaria de la Tabla Evento_X_SAlimentos_Intervalo
Id_Evento	int(10)	No		Indice hacia la tabla Eventos
Id_SAlimentos	smallint(5)	No		Indice hacia la tabla Cat_SAlimentos
Cantidad	smallint(5)	Yes	NULL	Cantidad de lugares para los alimentos
Fecha	date	No		Fecha de Evento
H_Inicio	time	No		Hora de inicio
H_Fin	time	No		Hora de finalización

Índice de la Tabla: "Evento_X_SAlimentos_Intervalo"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_ExSA	10	A		

Nombre de la Tabla: "Evento_X_SVarios"

Column	Type	Null	Default	Comments
Id_Evento	int(10)	No		Indice hacia la tabla Eventos
Id_SVarios	smallint(5)	No		Indice hacia la tabla Cat_ServiciosVarios
Fecha	date	No		Fecha de evento
SubTotal	decimal(10,2)	Yes	NULL	Subtotal de Pago

Índice de la Tabla: "Evento_X_SVarios"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Evento	0	A		
				Id_SVarios	0	A		
				Fecha	7	A		

Nombre de la Tabla: "Evento_X_SVarios_Intervalo"

Column	Type	Null	Default	Comments
Id_ExSV	bigint(20)	No		Llave Primaria de la Tabla Evento_X_SVarios_Intervalo
Id_Evento	int(10)	No		Indice hacia la Tabla Eventos
Id_SVarios	smallint(5)	No		Indice hacia la Tabla Cat_ServiciosVarios
Id_Espacio	smallint(5)	No	0	Indice hacia la Tabla Cat_Espacios
Cantidad	smallint(5)	Yes	NULL	Cantidad de personas
Fecha	date	No		Fecha del evento
H_Inicio	time	No		Hora de Inicio
H_Fin	time	No		Hora de Finalización

Índice de la Tabla: "Evento_X_SVarios_Intervalo"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_ExSV	7	A		

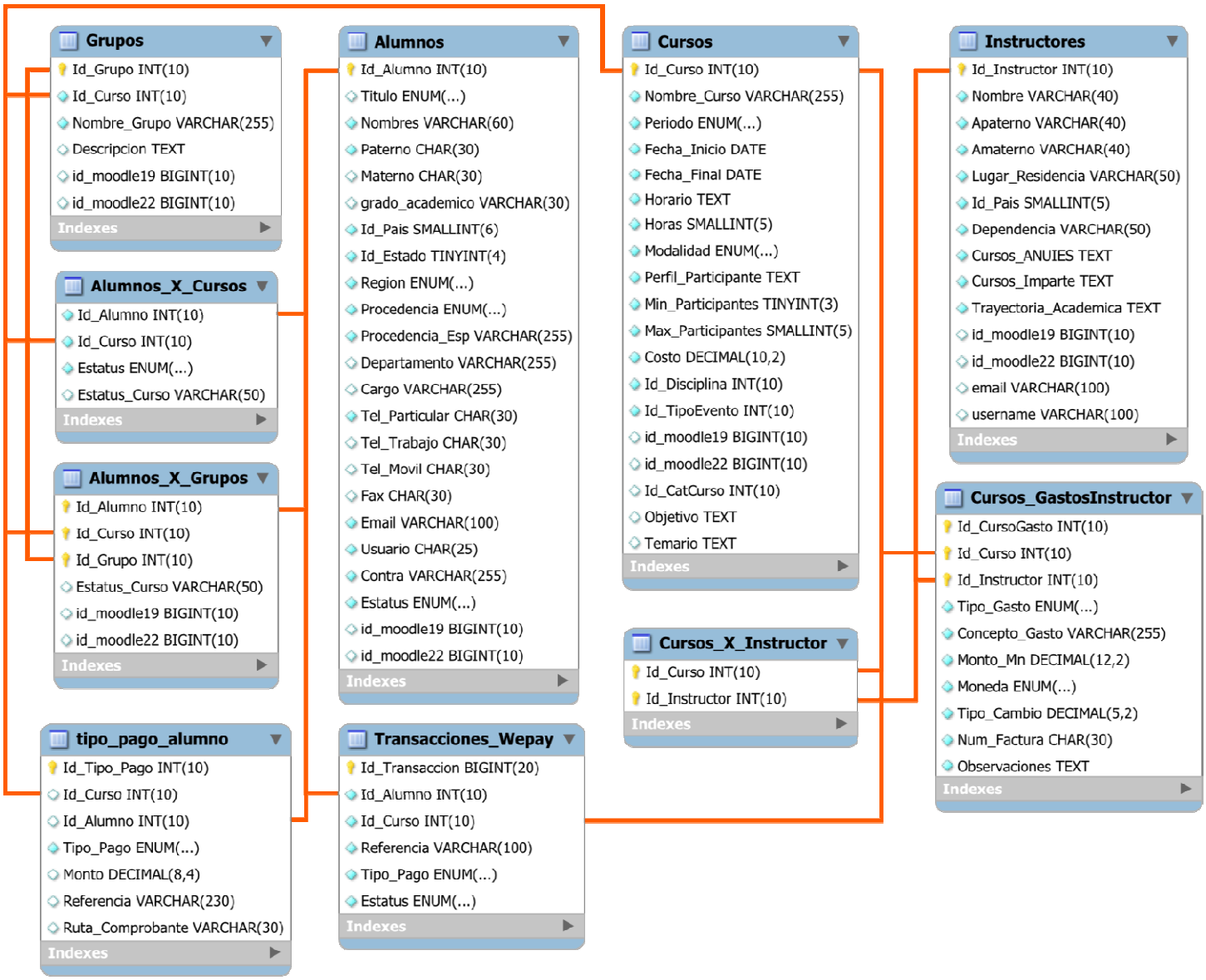
Nombre de la Tabla: "Evento_X_TipoEvento"

Column	Type	Null	Default	Comments
Id_Evento	smallint(5)	No		Indice hacia la Tabla Eventos
Id_TipoEvento	smallint(5)	No		Indice hacia la Tabla Cat_TiposEvento

Índice de la Tabla: "Evento_X_TipoEvento"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Evento	0	A		
				Id_TipoEvento	3	A		

Tabla utilizada por el módulo de sección académica



Nombre de la Tabla: "Instructores"

Column	Type	Null	Default	Comments
Id_Instructor	int(10)	No		Llave primaria de tabla instructores
Nombre	varchar(40)	No		Nombre del instructor
Apaterno	varchar(40)	No		Apellido Paterno del instructor
Amaterno	varchar(40)	No		Apellido Materno del instructor
Lugar_Residencia	varchar(50)	No		Lugar donde reside el instructor
Id_Pais	smallint(5)	No		Llave primaria del catalogo de paises
Dependencia	varchar(50)	No		Depencia o Institucion de procedencia del instructor
Cursos_ANUIES	text	No		Cursos que ha impartido en ANUIES
Cursos_Imparte	text	No		Cursos que imparte en general
Trayectoria_Academica	text	No		Descripción de la trayectoria académica del instructor
id_moodle19	bigint(10)	Yes	NULL	Id del Alumno en el mo0odle version 1.9
id_moodle22	bigint(10)	Yes	NULL	Id del Alumno en el mo0odle version 2.2
email	varchar(100)	Yes	NULL	Correo electrónico
username	varchar(100)	Yes	NULL	Nombre de usuario

Índice de la Tabla: "Instructores"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Instructor	5	A		
Id_Pais	BTREE	No	No	Id_Pais	0	A		

Nombre de la Tabla: "Cursos"

Column	Type	Null	Default	Comments
Id_Curso	int(10)	No		Llave Primaria de la Tabla Cursos
Nombre_Curso	varchar(255)	No		Nombre del Curso a iniciar.
Periodo	enum('Abierto', 'Mensual', 'Bimestral', 'Trimestral', 'Cuatrimestral', 'Semestral', 'Anual')	No	Abierto	Periodo de impartición.
Fecha_Inicio	date	No		Fecha de Inicio
Fecha_Final	date	No		Fecha de Finalización
Horario	text	No		Horario en que se impartirá
Horas	smallint(5)	No		Horas totales del duración
Modalidad	enum('Presencial', 'Distancia')	No		Modalidad
Perfil_Participante	text	No		Perfil de los participantes que tomaran el curso
Min_Participantes	tinyint(3)	No	7	Número de participantes mínimo para iniciar el curso
Max_Participantes	smallint(5)	No		Número de participantes total permitido
Costo	decimal(10,2)	No		Costo unitario por participante
Id_Disciplina	int(10)	No		Disciplina que cubrirá el curso Índice al catálogo de disciplinas
Id_TipoEvento	int(10)	No		Tipo de Evento Índice al Catalogo de Tipos de Evento
id_moodle19	bigint(10)	Yes	NULL	Id del Alumno en el Moodle version

				1.9
id_moodle22	bigint(10)	Yes	NULL	Id del Alumno en el Moodle version 2.2
Id_CatCurso	int(10)	Yes	NULL	Índice hacia la Tabla cat_cursos
Objetivo	text	Yes	NULL	Objetivo del Curso
Temario	text	Yes	NULL	Temario que se impartirá

Índice de la Tabla: "Cursos"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Curso	10	A		
Id_Disciplina	BTREE	No	No	Id_Disciplina	0	A		
				Id_TipoEvento	0	A		

Nombre de la Tabla: "Alumnos_X_Cursos"

Column	Type	Null	Default	Comments
Id_Alumno	int(10)	No		Llave Primaria de la Tabla Alumnos_X_Cursos
Id_Curso	int(10)	No		Índice hacia la tabla Cursos
Estatus	enum('Preinscrito', 'Inscrito')	No	Preinscrito	Estatus en que se encuentra el Alumno con respecto al curso
Estatus_Curso	varchar(50)	Yes	NULL	Estatus del Curso

Índice de la Tabla: "**Alumnos_X_Cursos"**

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Id_Alumno	BTREE	No	No	Id_Alumno	0	A		
				Id_Curso	0	A		

Nombre de la Tabla: "Alumnos_X_Grupos"

Column	Type	Null	Default	Comments
Id_Alumno	int(10)	No		Llave Primaria de la Tabla Alumnos_X_Grupos
Id_Curso	int(10)	No		Indice hacia la tabla Cursos
Id_Grupo	int(10)	No		Indice hacia la tabla Grupos
Estatus_Curso	varchar(50)	Yes	NULL	Estatus en que está encuentra el curso
id_moodle19	bigint(10)	Yes	NULL	Id del Alumno en el mo0odle version 1.9
id_moodle22	bigint(10)	Yes	NULL	Id del Alumno en el mo0odle version 2.2

Índice de la Tabla: "Alumnos_X_Grupos"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Alumno	0	A		
				Id_Curso	0	A		
				Id_Grupo	1	A		

Nombre de la Tabla: "Cursos_GastosInstructor"

Column	Type	Null	Default	Comments
Id_CursoGasto	int(10)	No		Llave Primaria de la Tabla Cursos_GastosInstructor
Id_Curso	int(10)	No		Índice hacia la tabla cat_cursos
Id_Instructor	int(10)	No		Índice hacia la tabla Cursos_X_Instructor
Tipo_Gasto	enum('Honorarios', 'Transporte', 'Alimentos', 'Hospedaje', 'Extras')	No		Tipo de gasto que se cubrirá
Concepto_Gasto	varchar(255)	No		Concepto de Gasto
Monto_Mn	decimal(12,2)	No		Monto a pagar
Moneda	enum('Pesos MN', 'US DLLS', 'Euros')	No		Tipo de moneda
Tipo_Cambio	decimal(5,2)	No		Tipo de cambio
Num_Factura	char(30)	No		Numero de factura generada
Observaciones	text	No		Observaciones

Índice de la Tabla: "Cursos_GastosInstructor"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_CursoGasto	0	A		
				Id_Curso	0	A		
				Id_Instructor	5	A		

Nombre de la Tabla: "Cursos_X_Instructor"

Column	Type	Null	Default	Comments
Id_Curso	int(10)	No		Indice hacia la tabla Cursos
Id_Instructor	int(10)	No		Indice hacia la tabla Instructores

Índice de la Tabla: "Cursos_X_Instructor"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Curso	0	A		
				Id_Instructor	6	A		

Nombre de la Tabla: "Alumnos"

Column	Type	Null	Default	Comments
Id_Alumno	int(10)	No		Llave Primaria de la Tabla Alumnos.
Titulo	enum('Lic.', 'Ing.', 'Mtra.', 'Mtro.', 'Dra.', 'Dr.')	Yes	NULL	Titulo del alumno
Nombres	varchar(60)	No		Nombres del Alumno.
Paterno	char(30)	No		Apellido Paterno del Alumno.
Materno	char(30)	Yes	NULL	Apellido Materno del Alumno.
grado_academico	varchar(30)	Yes	NULL	Grado Académico al que pertenece el Alumno

Id_Pais	smallint(6)	No		País de procedencia del Alumno.
Id_Estado	tinyint(4)	No		Estado de procedencia.
Region	enum('Centro-Occidente', 'Centro-Sur', 'Metropolitana', 'Noreste', 'Noroeste', 'Sur-Sureste')	No		Región de procedencia.
Procedencia	enum('Institución', 'Empresa', 'Independiente')	No		Si el Alumno procede de una Institución, Sociedad o es Independiente.
Procedencia_Esp	varchar(255)	No		Descripción ampliada de la procedencia.
Departamento	varchar(255)	Yes	NULL	Departamento laboral o área académica.
Cargo	varchar(255)	Yes	NULL	Cargo en la Institución o Sociedad del Alumno.
Tel_Particular	char(30)	No		Número telefónico particular.
Tel_Trabajo	char(30)	Yes	NULL	Número telefónico del trabajo.
Tel_Movil	char(30)	Yes	NULL	Número del móvil.
Fax	char(30)	Yes	NULL	Número del fax.
Email	varchar(100)	No		Correo electrónico del Alumno.
Usuario	char(25)	No		Nombre de Usuario.
Contra	varchar(255)	No		Contraseña.
Estatus	enum('Preinscrito', 'Inscrito')	No		Si el Alumno está Inscrito o Preinscrito.
id_moodle19	bigint(10)	Yes	NULL	Id del alumno en el moodle version 1.9
id_moodle22	bigint(10)	Yes	NULL	Id del alumno en el moodle version 2.2

Índices de la Tabla: “Alumnos”

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Alumno	6	A		

Nombre de la Tabla: “tipo_pago_alumno”

Column	Type	Null	Default	Comments
Id_Tipo_Pago	int(10)	No		Llave Primaria de la Tabla de los tipos de pago.
Id_Curso	int(10)	Yes	NULL	EL ide del curso al que pertenece el tipo de pago.
Id_Alumno	int(10)	Yes	NULL	EL ide del curso al que pertenece el tipo de pago.
Tipo_Pago	enum('Efectivo', 'Deposito', 'Webpay', 'Otro')	No		Los tipos de pagos aceptados
Monto	decimal(8,4)	Yes	NULL	El monto del pago, es opcional
Referencia	varchar(230)	Yes	NULL	Es la referencia al pagar por webpay
Ruta_Comprobante	varchar(30)	Yes	NULL	Es la ruta de donde quedo alojado el archivo

Índice de la Tabla: “tipo_pago_alumno”

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Tipo_Pago	7	A		

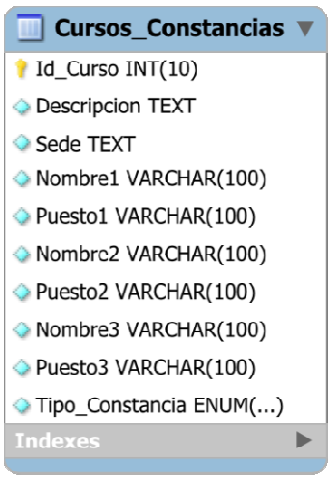
Nombre de la Tabla: "Transacciones_Wepay"

Column	Type	Null	Default	Comments
Id_Transaccion	bigint(20)	No		Llave Primaria de la Tabla Transacciones_Wepay
Id_Alumno	int(10)	No		Indice hacia la tabla Alumnos
Id_Curso	int(10)	No		Indice hacia la tabla Cursos
Estatus	enum('Intento de pago', 'Pagado', 'Error en Pago')	No		Estatus del pago

Índice de la Tabla: "Transacciones_Wepay"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Transaccion	2	A		

Tabla utilizada por el módulo de Reportes



Nombre de la Tabla: "Cursos_Constancias"

Column	Type	Null	Default	Comments
Id_Curso	int(10)	No		Indice hacia la tabla Cursos
Descripcion	text	No		Descripcion de la constancia
Sede	text	No		sede del curso
Nombre1	varchar(100)	No		Nombre de el titular
Puesto1	varchar(100)	No		Puesto de el titular
Nombre2	varchar(100)	No		Nombre de el titular
Puesto2	varchar(100)	No		Puesto de el titular
Nombre3	varchar(100)	No		Nombre de el titular
Puesto3	varchar(100)	No		Puesto de el titular
Tipo_Constancia	enum('Constancia', 'Diploma', 'Reconocimiento')	No		tipo de constancia

Índice de la Tabla: "Cursos_Constancias"

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	Id_Curso	0	A		

SEGURIDAD INCLUIDA EN EL SISTEMA MEDIANTE PRÁCTICAS DE PROGRAMACIÓN

- **Input Validation:**

√Conduct all data validation on a trusted system (e.g., The server)
Identify all data sources and classify them into trusted and untrusted. Validate all data from untrusted sources (e.g., Databases, file streams, etc.)
There should be a centralized input validation routine for the application
Specify proper character sets, such as UTF-8, for all sources of input
Encode data to a common character set before validating (Canonicalize)
All validation failures should result in input rejection
Determine if the system supports UTF-8 extended character sets and if so, validate after UTF-8 decoding is completed
Validate all client provided data before processing, including all parameters, URLs and HTTP header content (e.g. Cookie names and values). Be sure to include automated post backs from JavaScript, Flash or other embedded code
Verify that header values in both requests and responses contain only ASCII characters
Validate data from redirects (An attacker may submit malicious content directly to the target of the redirect, thus circumventing application logic and any validation performed before the redirect)
Validate for expected data types
Validate data range
Validate data length
Validate all input against a "white" list of allowed characters, whenever possible
If any potentially hazardous characters must be allowed as input, be sure that you implement additional controls like output encoding, secure task specific APIs and accounting for the utilization of that data throughout the application . Examples of common hazardous characters include:
<> " ' % ( ) & + \ \' \"
If your standard validation routine cannot address the following inputs, then they should be checked discretely
  - Check for null bytes (%00)
  - Check for new line characters (%0d, %0a, \r, \n)
  - Check for “dot-dot-slash" (../ or ..\) path alterations characters. In cases where UTF-8 extended character set encoding is supported, address alternate representation like: %c0%ae%c0%ae/ (Utilize canonicalization to address double encoding or other forms of obfuscation attacks)

- **Output Encoding**:

Conduct all encoding on a trusted system (e.g., The server)
Utilize a standard, tested routine for each type of outbound encoding
Contextually output encode all data returned to the client that originated outside the application's trust boundary. HTML entity encoding is one example, but does not work in all cases
Encode all characters unless they are known to be safe for the intended interpreter
Contextually sanitize all output of un-trusted data to queries for SQL, XML, and LDAP
Sanitize all output of un-trusted data to operating system commands

- **Authentication and Password Management:**

Require authentication for all pages and resources, except those specifically intended to be public
All authentication controls must be enforced on a trusted system (e.g., The server)
Establish and utilize standard, tested, authentication services whenever possible
Use a centralized implementation for all authentication controls, including libraries that call external authentication services
Segregate authentication logic from the resource being requested and use redirection to and from the centralized authentication control
All authentication controls should fail securely
All administrative and account management functions must be at least as secure as the primary authentication mechanism
If your application manages a credential store, it should ensure that only cryptographically strong one-way salted hashes of passwords are stored and that the table/file that stores the passwords and keys is write-able only by the application. (Do not use the MD5 algorithm if it can be avoided)
Password hashing must be implemented on a trusted system (e.g., The server).
Validate the authentication data only on completion of all data input, especially for sequential authentication implementations
Authentication failure responses should not indicate which part of the authentication data was incorrect. For example, instead of "Invalid username" or "Invalid password", just use "Invalid username and/or password" for both. Error responses must be truly identical in both display and source code
Utilize authentication for connections to external systems that involve sensitive information or functions
Authentication credentials for accessing services external to the application should be encrypted and stored in a protected location on a trusted system (e.g., The server). The source code is NOT a secure location
Use only HTTP POST requests to transmit authentication credentials
Only send non-temporary passwords over an encrypted connection or as encrypted data, such as in an encrypted email. Temporary passwords associated with email resets may be an exception
Enforce password complexity requirements established by policy or regulation. Authentication credentials should be sufficient to withstand attacks that are typical of the threats in the deployed environment. (e.g., requiring the use of alphabetic as well as numeric and/or special characters)
Enforce password length requirements established by policy or regulation. Eight characters is commonly used, but 16 is better or consider the use of multi-word pass phrases
Password entry should be obscured on the user's screen. (e.g., on web forms use the input type "password")
Enforce account disabling after an established number of invalid login attempts (e.g., five attempts is common). The account must be disabled for a period of time sufficient to discourage brute force guessing of credentials, but not so long as to allow for a denial-of-service attack to be performed
Password reset and changing operations require the same level of controls as account creation and authentication.
Password reset questions should support sufficiently random answers. (e.g., "favorite book" is a bad question because “The Bible” is a very common answer)
If using email based resets, only send email to a pre-registered address with a temporary link/password
Temporary passwords and links should have a short expiration time
Enforce the changing of temporary passwords on the next use
Notify users when a password reset occurs
Prevent password re-use
Passwords should be at least one day old before they can be changed, to prevent attacks on password re-use
Enforce password changes based on requirements established in policy or regulation. Critical systems may require more frequent changes. The time between resets must be administratively controlled
Disable "remember me" functionality for password fields

The last use (successful or unsuccessful) of a user account should be reported to the user at their next successful login
Implement monitoring to identify attacks against multiple user accounts, utilizing the same password. This attack pattern is used to bypass standard lockouts, when user IDs can be harvested or guessed
Change all vendor-supplied default passwords and user IDs or disable the associated accounts
Re-authenticate users prior to performing critical operations
Use Multi-Factor Authentication for highly sensitive or high value transactional accounts
If using third party code for authentication, inspect the code carefully to ensure it is not affected by any malicious code

- **Session Management:**

Use the server or framework's session management controls. The application should only recognize these session identifiers as valid
Session identifier creation must always be done on a trusted system (e.g., The server)
Session management controls should use well vetted algorithms that ensure sufficiently random session identifiers
Set the domain and path for cookies containing authenticated session identifiers to an appropriately restricted value for the site
Logout functionality should fully terminate the associated session or connection
Logout functionality should be available from all pages protected by authorization
Establish a session inactivity timeout that is as short as possible, based on balancing risk and business functional requirements. In most cases it should be no more than several hours
Disallow persistent logins and enforce periodic session terminations, even when the session is active. Especially for applications supporting rich network connections or connecting to critical systems. Termination times should support business requirements and the user should receive sufficient notification to mitigate negative impacts
If a session was established before login, close that session and establish a new session after a successful login
Generate a new session identifier on any re-authentication
Do not allow concurrent logins with the same user ID
Do not expose session identifiers in URLs, error messages or logs. Session identifiers should only be located in the HTTP cookie header. For example, do not pass session identifiers as GET parameters
Protect server side session data from unauthorized access, by other users of the server, by implementing appropriate access controls on the server
Generate a new session identifier and deactivate the old one periodically. (This can mitigate certain session hijacking scenarios where the original identifier was compromised)
Generate a new session identifier if the connection security changes from HTTP to HTTPS, as can occur during authentication. Within an application, it is recommended to consistently utilize HTTPS rather than switching between HTTP to HTTPS.
Supplement standard session management for sensitive server-side operations, like account management, by utilizing per-session strong random tokens or parameters. This method can be used to prevent Cross Site Request Forgery attacks
Supplement standard session management for highly sensitive or critical operations by utilizing per-request, as opposed to per-session, strong random tokens or parameters
Set the "secure" attribute for cookies transmitted over an TLS connection
Set cookies with the HttpOnly attribute, unless you specifically require client-side scripts within your application to read or set a cookie's value

- **Access Control:**

Use only trusted system objects, e.g. server side session objects, for making access authorization decisions
Use a single site-wide component to check access authorization. This includes libraries that call external authorization services
Access controls should fail securely
Deny all access if the application cannot access its security configuration information
Enforce authorization controls on every request, including those made by server side scripts, "includes" and requests from rich client-side technologies like AJAX and Flash
Segregate privileged logic from other application code
Restrict access to files or other resources, including those outside the application's direct control, to only authorized users
Restrict access to protected URLs to only authorized users
Restrict access to protected functions to only authorized users
Restrict direct object references to only authorized users
Restrict access to services to only authorized users
Restrict access to application data to only authorized users
Restrict access to user and data attributes and policy information used by access controls
Restrict access security-relevant configuration information to only authorized users
Server side implementation and presentation layer representations of access control rules must match
If state data must be stored on the client, use encryption and integrity checking on the server side to catch state tampering.
Enforce application logic flows to comply with business rules
Limit the number of transactions a single user or device can perform in a given period of time. The transactions/time should be above the actual business requirement, but low enough to deter automated attacks
Use the "referer" header as a supplemental check only, it should never be the sole authorization check, as it is can be spoofed
If long authenticated sessions are allowed, periodically re-validate a user's authorization to ensure that their privileges have not changed and if they have, log the user out and force them to re-authenticate
Implement account auditing and enforce the disabling of unused accounts (e.g., After no more than 30 days from the expiration of an account's password.)
The application must support disabling of accounts and terminating sessions when authorization ceases (e.g., Changes to role, employment status, business process, etc.)
Service accounts or accounts supporting connections to or from external systems should have the least privilege possible
Create an Access Control Policy to document an application's business rules, data types and access authorization criteria and/or processes so that access can be properly provisioned and controlled. This includes identifying access requirements for both the data and system resources

- **Cryptographic Practices:**

All cryptographic functions used to protect secrets from the application user must be implemented on a trusted system (e.g., The server)
Protect master secrets from unauthorized access
Cryptographic modules should fail securely
All random numbers, random file names, random GUIDs, and random strings should be generated using the cryptographic module's approved random number generator when these random values are intended to be un-guessable
Cryptographic modules used by the application should be compliant to FIPS 140-2 or an equivalent standard. (See http://csrc.nist.gov/groups/STM/cmvp/validation.html)

Establish and utilize a policy and process for how cryptographic keys will be managed

- **Error Handling and Logging:**

Do not disclose sensitive information in error responses, including system details, session identifiers or account information
Use error handlers that do not display debugging or stack trace information
Implement generic error messages and use custom error pages
The application should handle application errors and not rely on the server configuration
Properly free allocated memory when error conditions occur
Error handling logic associated with security controls should deny access by default
All logging controls should be implemented on a trusted system (e.g., The server)
Logging controls should support both success and failure of specified security events
Ensure logs contain important log event data
Ensure log entries that include un-trusted data will not execute as code in the intended log viewing interface or software
Restrict access to logs to only authorized individuals
Utilize a master routine for all logging operations
Do not store sensitive information in logs, including unnecessary system details, session identifiers or passwords
Ensure that a mechanism exists to conduct log analysis
Log all input validation failures
Log all authentication attempts, especially failures
Log all access control failures
Log all apparent tampering events, including unexpected changes to state data
Log attempts to connect with invalid or expired session tokens
Log all system exceptions
Log all administrative functions, including changes to the security configuration settings
Log all backend TLS connection failures
Log cryptographic module failures
Use a cryptographic hash function to validate log entry integrity

- **Data Protection:**

Implement least privilege, restrict users to only the functionality, data and system information that is required to perform their tasks
Protect all cached or temporary copies of sensitive data stored on the server from unauthorized access and purge those temporary working files a soon as they are no longer required.
Encrypt highly sensitive stored information, like authentication verification data, even on the server side. Always use well vetted algorithms, see "Cryptographic Practices" for additional guidance
Protect server-side source-code from being downloaded by a user
Do not store passwords, connection strings or other sensitive information in clear text or in any non-cryptographically secure manner on the client side. This includes embedding in insecure formats like: MS viewstate, Adobe flash or compiled code
Remove comments in user accessible production code that may reveal backend system or other sensitive information
Remove unnecessary application and system documentation as this can reveal useful information to attackers
Do not include sensitive information in HTTP GET request parameters
Disable auto complete features on forms expected to contain sensitive information, including authentication

Disable client side caching on pages containing sensitive information. Cache-Control: no-store, may be used in conjunction with the HTTP header control "Pragma: no-cache", which is less effective, but is HTTP/1.0 backward compatible
The application should support the removal of sensitive data when that data is no longer required. (e.g. personal information or certain financial data)
Implement appropriate access controls for sensitive data stored on the server. This includes cached data, temporary files and data that should be accessible only by specific system users

- **Communication Security:**

Implement encryption for the transmission of all sensitive information. This should include TLS for protecting the connection and may be supplemented by discrete encryption of sensitive files or non-HTTP based connections
TLS certificates should be valid and have the correct domain name, not be expired, and be installed with intermediate certificates when required
Failed TLS connections should not fall back to an insecure connection
Utilize TLS connections for all content requiring authenticated access and for all other sensitive information
Utilize TLS for connections to external systems that involve sensitive information or functions
Utilize a single standard TLS implementation that is configured appropriately
Specify character encodings for all connections
Filter parameters containing sensitive information from the HTTP referer, when linking to external sites

- **System Configuration:**

Ensure servers, frameworks and system components are running the latest approved version
Ensure servers, frameworks and system components have all patches issued for the version in use
Turn off directory listings
Restrict the web server, process and service accounts to the least privileges possible
When exceptions occur, fail securely
Remove all unnecessary functionality and files
Remove test code or any functionality not intended for production, prior to deployment
Prevent disclosure of your directory structure in the robots.txt file by placing directories not intended for public indexing into an isolated parent directory. Then "Disallow" that entire parent directory in the robots.txt file rather than Disallowing each individual directory
Define which HTTP methods, Get or Post, the application will support and whether it will be handled differently in different pages in the application
Disable unnecessary HTTP methods, such as WebDAV extensions. If an extended HTTP method that supports file handling is required, utilize a well-vetted authentication mechanism
If the web server handles both HTTP 1.0 and 1.1, ensure that both are configured in a similar manor or insure that you understand any difference that may exist (e.g. handling of extended HTTP methods)
Remove unnecessary information from HTTP response headers related to the OS, web-server version and application frameworks
The security configuration store for the application should be able to be output in human readable form to support auditing
Implement an asset management system and register system components and software in it
Isolate development environments from the production network and provide access only to authorized development and test groups. Development environments are often configured less securely than production environments and attackers may use this difference to discover shared weaknesses or as an avenue for exploitation

Implement a software change control system to manage and record changes to the code both in development and production

- **Database Security:**

Use strongly typed parameterized queries
Utilize input validation and output encoding and be sure to address meta characters. If these fail, do not run the database command
Ensure that variables are strongly typed
The application should use the lowest possible level of privilege when accessing the database
Use secure credentials for database access
Connection strings should not be hard coded within the application. Connection strings should be stored in a separate configuration file on a trusted system and they should be encrypted.
Use stored procedures to abstract data access and allow for the removal of permissions to the base tables in the database
Close the connection as soon as possible
Remove or change all default database administrative passwords. Utilize strong passwords/phrases or implement multi-factor authentication
Turn off all unnecessary database functionality (e.g., unnecessary stored procedures or services, utility packages, install only the minimum set of features and options required (surface area reduction))
Remove unnecessary default vendor content (e.g., sample schemas)
Disable any default accounts that are not required to support business requirements
The application should connect to the database with different credentials for every trust distinction (e.g., user, read-only user, guest, administrators)

- **File Management:**

Do not pass user supplied data directly to any dynamic include function
Require authentication before allowing a file to be uploaded
Limit the type of files that can be uploaded to only those types that are needed for business purposes
Validate uploaded files are the expected type by checking file headers. Checking for file type by extension alone is not sufficient
Do not save files in the same web context as the application. Files should either go to the content server or in the database.
Prevent or restrict the uploading of any file that may be interpreted by the web server.
Turn off execution privileges on file upload directories
Implement safe uploading in UNIX by mounting the targeted file directory as a logical drive using the associated path or the chrooted environment
When referencing existing files, use a white list of allowed file names and types. Validate the value of the parameter being passed and if it does not match one of the expected values, either reject it or use a hard coded default file value for the content instead
Do not pass user supplied data into a dynamic redirect. If this must be allowed, then the redirect should accept only validated, relative path URLs
Do not pass directory or file paths, use index values mapped to pre-defined list of paths
Never send the absolute file path to the client
Ensure application files and resources are read-only
Scan user uploaded files for viruses and malware

- **Memory Management:**

Utilize input and output control for un-trusted data
Double check that the buffer is as large as specified
When using functions that accept a number of bytes to copy, such as strncpy(), be aware that if the destination buffer size is equal to the source buffer size, it may not NULL-terminate the string
Check buffer boundaries if calling the function in a loop and make sure there is no danger of writing past the allocated space
Truncate all input strings to a reasonable length before passing them to the copy and concatenation functions
Specifically close resources, don't rely on garbage collection. (e.g., connection objects, file handles, etc.)
Use non-executable stacks when available
Avoid the use of known vulnerable functions (e.g., printf, strcat, strcpy etc.)
Properly free allocated memory upon the completion of functions and at all exit points

- **General Coding Practices:**

Use tested and approved managed code rather than creating new unmanaged code for common tasks
Utilize task specific built-in APIs to conduct operating system tasks. Do not allow the application to issue commands directly to the Operating System, especially through the use of application initiated command shells
Use checksums or hashes to verify the integrity of interpreted code, libraries, executables, and configuration files
Utilize locking to prevent multiple simultaneous requests or use a synchronization mechanism to prevent race conditions
Protect shared variables and resources from inappropriate concurrent access
Explicitly initialize all your variables and other data stores, either during declaration or just before the first usage
In cases where the application must run with elevated privileges, raise privileges as late as possible, and drop them as soon as possible
Avoid calculation errors by understanding your programming language's underlying representation and how it interacts with numeric calculation. Pay close attention to byte size discrepancies, precision, signed/unsigned distinctions, truncation, conversion and casting between types, "not-a-number" calculations, and how your language handles numbers that are too large or too small for its underlying representation
Do not pass user supplied data to any dynamic execution function
Restrict users from generating new code or altering existing code
Review all secondary applications, third party code and libraries to determine business necessity and validate safe functionality, as these can introduce new vulnerabilities
Implement safe updating. If the application will utilize automatic updates, then use cryptographic signatures for your code and ensure your download clients verify those signatures. Use encrypted channels to transfer the code from the host server.

G L O S A R I O

APC (PHP libreria)

Por sus siglas Alternative PHP Cache, es una librería para monitorear el rendimiento o avances de PHP, además de generar un cache alternativo mas optimizado al tradicional.

Background

Fondo, en este caso puede usarse una imagen como patrón o un color.

Browser (Navegador)

Es una aplicación con la capacidad de interpretar el código HTML, archivos y scripts de las páginas web, las cuales pueden encontrarse hospedadas en servidores dentro de la World Wide Web o localmente.

CSS (Cascading Style Sheet) (Hojas de estilos en cascada)

Este tipo de scripts permite separar el contenido del diseño en una página web, por medio de selectores, propiedades y atributos, permite manipular y controlar de manera estratégia aspectos como lo son; tipografías, diseño, colores, posición de elementos en el layout, etc.

CSV (Comma-separated values)

Se trata de archivos con cualquier cantidad de registros en formato de texto simple, los cuales guardan datos tabulares (númericos y texto) separados por comas.

Dirección IP

Es un código numérico que etiqueta e identifica a un equipo de cómputo o dispositivo participante dentro de una red y que utiliza para comunicarse el Protocolo de Internet (IP) y nos dice de éste donde se encuentra ubicado.

Document Root (Raíz de documentos)

Es la carpeta a nivel raíz donde se encuentran todos los archivos y carpetas con scripts, código y material con los que opera el sistema.

Engine

En este caso se refiere a la database engine o motor de almacenamiento y es el componente de software detrás de un manejador de bases de datos (DBMS) Database Management System el DBMS utiliza el engine para crear, leer, actualizar y eliminar datos de la base (p. ej. MyISAM es un tipo de engine así como InnoDB, etc.).

Estilos inline

Instrucciones de estilos en cascada (CSS) aplicadas de manera selectiva, directa y específica a las etiquetas HTML de un documento o archivo web.

Framework

Un framework de software se refiere a una plataforma universal para el desarrollo de aplicaciones, productos y soluciones. Estos frameworks pueden constar de aplicaciones de apoyo, compiladores, librerías de código, interfases de programación y conjuntos de herramientas que agrupan todos los distintos componentes para habilitar el desarrollo de un proyecto o solución.

GET

Es un de varios métodos de requerimiento soportados por el protocolo HTTP utilizado por la World Wide Web. Solicita una representación del recurso especificado. Adicionalmente solicitudes realizadas a través de este método deben únicamente obtener información y de ninguna manera otro tipo de efecto.

GIF (Graphics Interchange Format) (Formato de intercambio de gráficos)

Es un formato de imagen matricial o de mapa de bits, bitmap o imagen ráster, definida por su dimensión en pixeles (ancho y alto) y por profundidad de color, es decir la cantidad de colores, tonos o matices que pueden almacenarse por cada punto o pixel.

HTML (HyperText Markup Language) (Lenguage de marcado de hipertexto)

Es la parte más básica de las páginas web y hoy día estas etiquetas definen primordialmente la composición estructural de los documentos web por medio de un conjunto de elementos o etiquetas claramente definidas y utilizadas para la formación del contenido.

GUI (Graphical User Interface) (Interfaz gráfica de usuario)

Utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz, proporciona un entorno visual de manera intuitiva y para permitir la conexión física y funcional entre dos sistemas o dispositivos de cualquier tipo permitiendo una comunicación interactiva entre distintos niveles.

JavaScript

Lenguaje de programación dinámica orientado a objetos, utilizado para mejorar la experiencia del usuario por medio del navegador.

JPEG (Joint Photographic Experts Group)

Las siglas se refieren al comité que creó el estándar. Es un método de compresión de imagen para fotografías digitales, consta de una escala de compresión que va del 1 al 10; donde 1 equivale a la calidad y espacio de almacenamiento menores y por lo tanto 10 equivale al máximo.

JSON (JavaScript Object Notation) (Notación literal de objetos de JavaScript)

Es un formato ligero para el intercambio de datos que no requiere el uso de XML y es empleado de manera habitual en entornos donde el tamaño del flujo de datos entre cliente y servidor es de vital importancia cuando la fuente de datos es explícitamente confiable; actualmente Yahoo y Google son ejemplos de su uso extensivo.

Layout (Acomodo)

Es la parte del Diseño Gráfico que se refiere a la distribución o acomodo de los elementos o contenidos, en este caso de una página web.

LMS

Un sistema de gestión de aprendizaje es un software instalado en un servidor web que se emplea para administrar, distribuir y controlar las actividades de formación no presencial (o aprendizaje electrónico) de una institución u organización.

Librerías gráficas GD (Graphics Draw) (dibujado de gráficos)

Es una aplicación para gestionar la manipulación de imágenes y cuenta con la capacidad para generar formatos como lo son GIF, JPEG, PNG entre otros.

Moodle

es una aplicación web de tipo Ambiente Educativo Virtual, un sistema de gestión de cursos, de distribución libre, que ayuda a los educadores a crear comunidades de aprendizaje en línea. Este tipo de plataformas tecnológicas también se conoce como LMS (Learning Management System).

Namespace (Espacio de nombre)

Se trata de un contenedor abstracto o ambiente creado para servir de soporte a agrupaciones lógicas de identificadores o símbolos únicos (nombres). De tal manera que no ocasiona problemas usar el mismo identificador para dos instancias independientes cada una en cualquiera de dos o más namespaces que les corresponda, esto por lo tanto permite al sistema ser más modular.

Open source (Código abierto)

Es el término por el que el software de libre distribución y desarrollo es conocido.

PNG (Portable Network Graphics) (Gráficos de red portátiles)

Es un formato de compresión de imagen sin pérdida para mapas de bits, rásters, etc. y fue desarrollado con el fin de mejorar, proveer de ventajas y eventualmente reemplazar al formato GIF.

POST

Es un de varios métodos de requerimiento soportados por el protocolo HTTP utilizado por la World Wide Web. Es usado cuando el cliente necesita enviar información al servidor como parte de la solicitud o requerimiento, como cuando se carga un archivo o se envía un formulario completado. Contrario al método GET, las solicitudes vía POST incluyen adicionalmente un mensaje, esto permite enviar al servidor una gran cantidad de información.

Progressive enhancement (Mejoramiento progresivo)

El mejoramiento progresivo ha sido adoptado por desarrolladores profesionales como una norma que cumpla con los estándares y las buenas prácticas del Diseño Web preocupado por la accesibilidad, semántica, métodos de aplicación de cascadas de estilo externas (CSS); por medio de estrategias tecnológicas y el uso

de JavaScript, de tal forma que permite que cualquiera tenga acceso al contenido y a la funcionalidad básica de una página web, usando cualquier navegador o conexión de Internet; al mismo tiempo que otros con disposición de mayores y más avanzados recursos obtengan el máximo provecho de la página.

RPC (Remote Procedure Call) (Llamada a procedimiento remoto)

Es un protocolo ampliamente utilizado en entornos cliente-servidor que permite que un programa o aplicación ejecute un código en otro equipo remoto sin necesidad de preocuparse por la comunicación existente entre ambos dispositivos, en este caso es el cliente quien inicia el proceso solicitando al servidor que ejecute un procedmiento en particular y el servidor en respuesta envía de regreso el resultado de dicha operación al cliente

Salt (sal)

En criptografía una Salt consiste de partes aleatorias y puede ser usada como parte de una clave en un cifrado o como algoritmo de encripción de contraseñas con el fin de complicar los ataques.

URL (Uniform Resource Locator) (Localizador de recursos uniforme)

Es un conjunto de caracteres que hace referencia a un recurso en la Internet.

W3C (World Wide Web Consortium) (Consorcio de la Internet Mundial)

El W3C o W3 es un consorcio internacional conformado por empresas trasnacionales de reconocimiento mundial en las Tecnologías de Información (TI) y es la encargada de genererar recomendaciones y establecer estándares y normas para la World Wide Web. Creada y dirigida por Tim Berners-Lee en el Instituto Tecnológico de Massachusetts (MIT), creador original de las principales tecnologías en las que la web está basada.

W3C Recommendation (Recomendación del W3C)

Una recomendación del W3C es la etapa final de un proceso de ratificación por parte del grupo de trabajo del Consorcio de la World Wide Web en lo concerniente a estándares técnicos. Esta designación significa que un documento ha sido sujeto a una revisión pública por la organización de miembros del W3C. Su objetivo está dirigido a estandarizar las tecnologías web y es el equivalente a estándares técnicos publicados por muchas otras industrias.

Widget

Es una especie de control o elemento interactivo parte de una interfaz gráfica de usuario que despliega arreglos de información modificable por el usuario por medio de una ventana o recuadro de texto.

WYSWYG (What You See, What You Get) (“Lo que ve, es lo que obtiene”)

Describe el funcionamiento de herramientas o aplicaciones cuyo objetivo es facilitar la formación de documentos web, aún cuando la persona que los utilice, posea básico o escaso conocimiento en la construcción semántica de contenido web conjuntamente con el uso de estilos (CSS) permitiendo al usuario no preocuparse demasiado por la forma en la cual el código está siendo generado detrás.